

2012

Autor:

Gonzalo Herranz Martínez de
Pinillos.

TFG: Trabajo de Fin de Grado.

Tutor del proyecto:

Javier Fernández Muñoz.
Alberto Jardón Huete.



Universidad Carlos III de Madrid.

**DISEÑO E INTEGRACIÓN EN
ANDROID DE UN MÓDULO DE
RECONOCIMIENTO DE VOZ PARA
UN SISTEMA DE DOMÓTICA
ASISTENCIAL.**

23 de Mayo del 2012.

Agradecimientos.

En un mundo globalizado y con la competitividad actual, agradecer a quienes hacen lo posible (de una u otra forma) para darte la oportunidad de prepararte y estudiar, es esencial. Se dice que una persona te da una oportunidad cuando el sacrificio de algo en su vida, supone una oportunidad para tí.

Pues bien, me siento un afortunado, al poder decir que mi familia me ha dado mil y una oportunidades con el único propósito de ayudarme a desarrollar. Las capacidades de una persona, pueden ser espectaculares e incluso únicas, no obstante no son todo lo que podrían ser, hasta que éstas, son cultivadas del modo adecuado y con el tiempo necesario. Evidentemente, todo esto no es sencillo, incluso en ocasiones, se puede tornar en complicado si no se cuenta con el tiempo y el apoyo económico de algo tan sólido como una familia.

Es por ello, que quiero agradecer a mi madre, todas las tutorías y tardes interminables, en las que se empeñaba en hacerme una persona más responsable y estudiosa. A mi padre, por todos aquellos momentos en los que, a pesar de que no estuviese presente, sentía su apoyo incondicional. Me enseñó que el trabajo y el esfuerzo sano, es siempre recompensado, y que cuando una persona pone el corazón en lo que hace, es imposible que salga mal. Por supuesto una mención especial a mi hermano mayor Jesús, el cual siempre me alejaba las metas para que yo me esforzase algo más en alcanzarlas, pero siempre con su constante supervisión. Quiero agradecer también, a figuras como mi abuela Carmen o mi tía-abuela M^a Ángeles su constante e incondicional apoyo a lo largo de tantos años, me alegra que ambas puedan ver cómo termino mis estudios superiores. Ellas son en parte las responsables de que todos nosotros estemos aquí.

Agradecer también a mi novia actual Patricia todo lo que ha tenido que esforzarse durante nuestros actuales cinco años de relación. Empecé mis estudios prácticamente a la vez que mi relación con ella y actualmente me ve finalizarla con la misma ilusión con la que se inició. Otro pilar fundamental de mi vida que ayudó a construir lo que es en la actualidad.

Las figuras que se podrían seguir enumerando son innumerables, mi Abuela Concepción, mi tío y padrino Luis, mis tías Julia y Cuqui o mi tío Javier, sin embargo seguro que se me olvidaría alguien y por eso quiero agradecer en general el apoyo de mi familia.

Finalmente cómo no, agradecer a todos los docentes que hasta el momento, se han hecho cargo de mi educación. Me gustaría poder nombrar a todos sin embargo me resulta imposible. No obstante, he de enumerar el Colegio San Patricio y la Universidad Carlos III como las dos instituciones que me han ayudado a formarme. No sólo de forma académica sino como persona y profesional. En el futuro siempre trataré de llevar con orgullo el nombre de ambos centros en mi currículum profesional y demostrar la preparación que ofrecen.

Sin todas estas personas (cada uno en su ámbito), yo, jamás hubiese llegado a ser quien soy en la actualidad. Y por supuesto, nunca podría llegar a ser la persona que espero ser, es por ello, que me podría extender de forma indefinida en éste apartado, dado que pocas palabras son capaces de expresar lo que siento. Me despido con una cita que me ha ayudado siempre mucho en la vida.

Errar es humano y como tal, hay que aceptarlo. A pesar de ello, se ha de mirar al futuro desde la perspectiva del presente y sin olvidar nunca el pasado.

Resumen.

En los últimos años, las grandes empresas de telefonía móvil han estado subvencionando terminales de última generación a sus clientes. Estos terminales, gozan de una capacidad de computación, de transferencia de banda ancha, y de especificaciones técnicas inimaginables hace unos años.

El primer 'Smartphone' lanzado a nivel comercial y con gran aceptación de los usuarios fue, el iPhone. Dicho terminal, supuso una revolución tecnológica en el concepto de terminal móvil que se tenía de forma popular. Tras el éxito a nivel mundial del terminal, y debido a la gran aceptación del usuario, se pudo observar el éxito que supondría este tipo de dispositivos.

A partir de dicho lanzamiento, muchos fueron los fabricantes que intentaron inundar el mercado con sus terminales. No obstante, fue HTC con su integración del hasta entonces desconocido, sistema operativo 'Android' el que consiguió dicha superioridad. La familiaridad con la que se adaptaron la gran parte de los desarrolladores a este nuevo sistema operativo basado en Linux, fue la clave del éxito del mismo. Debido a que se trata de un sistema operativo basado en "Linux" la gran parte de sus aplicaciones gestionadas en su 'Market' o actualmente 'Google Play' son de descarga e instalación gratuita. Por lo que se puede decir que desde el año 2010 'Android' se ha posicionado en el sistema operativo elegido por la gran mayoría de usuarios de teléfonos inteligentes a nivel mundial.

Debido a todo ello, surgen ideas, que se complementan con conocimientos, dando lugar aplicaciones que en ciertos casos son simplemente originales pero muy prácticas y otras sumamente costosas y poco prácticas. Adivinen cuál triunfa y cuál nunca se debería haber ni si quiera analizado.

A partir de esta idea, surgió la aplicación que en este proyecto se va a tratar de detallar de forma completa y adecuada. Es una aplicación, que cumple una labor social innegable. Y es que, se trata de una aplicación en la que una persona con un cierto impedimento físico, será capaz de valerse por sí misma dentro de su propio hogar.

Lo que se pretende desarrollar en este proyecto es un sistema domótico con aplicación móvil que sea capaz de satisfacer las necesidades de una persona discapacitada en el grado más avanzado posible.

Prólogo.

Un modulo de domótica asistencial, consta de una serie de elementos esenciales. La infraestructura requerida a nivel físico es innegable y el sistema de automatización, por supuesto también.

En este proyecto, se pretende abarcar el análisis de un sistema domótico en su sentido físico, y a partir del mismo, desarrollar una aplicación para el sistema operativo Android. Dicha aplicación, deberá de ser capaz de satisfacer los deseos de una persona de discapacidad avanzada. El sistema final, deberá reconocer objetos y acciones de modo auditivo y además debe poder integrar un módulo visual para el reconocimiento de elementos. El usuario, deberá focalizar su mirada en un electrodoméstico o sistema tecnológico previamente acondicionado para ello y a continuación lanzar un comando de voz. En este proyecto, nos centraremos más en el módulo de reconocimiento de voz.

Por tanto, se puede suponer que el sistema domótico deberá ser autónomo, de tal modo que cuando identifique el objeto según un código de barras y acto seguido, reciba el mandato auditivo, automatice la acción a realizar. Éste sistema, contendrá una serie de escenarios que deberán ser analizados y probados. Se considerará de mayor prioridad la adaptabilidad del sistema tanto a las condiciones de una persona discapacitada como a su lenguaje. El sistema será capaz de soportar cualquier idioma del dispositivo.

Índice de contenidos:

Agradecimientos.....	3
Resumen.....	5
Prólogo.....	7
Índice de contenidos:.....	9
Índice de Tablas:.....	13
Índice de Ilustraciones:	17
1. Introducción.....	19
1.1 Motivación.....	19
1.2 Objetivos.....	21
1.3 Contexto socio-cultural y económico.....	21
1.4 Estructura del documento.....	22
2. Estado del arte.....	25
2.1 Sistemas de Domótica.....	25
2.1.1 Introducción.....	25
2.1.2 Servicios en sistemas de domótica.....	26
2.1.2.1 Ahorro energético.....	26
2.1.2.2 Confort:	26
2.1.2.3 Seguridad.....	27
2.1.2.4 Comunicaciones	27
2.1.2.5 Accesibilidad.....	27
2.1.3 El sistema domótico	28
2.1.3.1 Dispositivos	28
2.1.3.2 Arquitectura	28
2.1.3.3 Elementos de una instalación domótica	29
2.1.4 Asociaciones Internacionales y Españolas de domótica	29
2.1.5 Hogar Digital.....	30
2.1.5.1 Diferencias entre casa domótica y hogar digital.....	30
2.2 Ley de dependencia (20 de Abril del 2006) ^[1]	31
2.2.1 Prestaciones y catálogo de servicios para personas dependientes ^[1]	31
2.2.2 Baremos de aplicación de la ley ^[1]	32
2.2.3 Aplicabilidad real de la ley y situación actual.....	32

2.3	Sistemas Móviles.....	33
2.3.1	Introducción	33
2.3.2	Dispositivos Móviles.....	34
2.3.3	Aplicaciones Móviles	35
2.3.4	Principales Sistemas Operativos Móviles.....	36
2.3.5	Sistema Operativo - Android ^[3]	39
2.3.5.1	Historia	40
2.3.5.2	Historial de Actualizaciones y Versiones ^[4]	41
2.3.5.3	Arquitectura:	46
2.3.5.4	Aplicaciones.....	47
2.4	Aplicaciones móviles de asistencia domótica.	48
2.4.1	Ejemplos de aplicaciones domóticas:.....	48
2.4.2	Competidores en la actualidad:	51
3.	Gestión del Proyecto.....	53
3.1	Introducción.	53
3.2	Metodología del proyecto.....	53
3.2.1	Fases de la metodología adoptada:	55
3.3	Ciclo de vida.	57
3.4	Organización de Planificación.	59
3.5	Planificación.	61
3.6	Presupuesto.	63
4.	Análisis del Sistema.....	67
4.1	Introducción.	67
4.2	Requisitos del Sistema.....	67
4.2.1	Requisitos de Usuario.....	68
4.2.1.1	Requisitos de Capacidad.	69
4.2.1.2	Requisitos de Restricción.	72
4.2.2	Requisitos del Software.....	73
4.2.2.1	Requisitos Funcionales.....	73
4.2.2.2	Requisitos No Funcionales.	81
4.2.3	Matriz de trazabilidad de requisitos de usuario y requisitos del software.	83
4.3	Casos de Uso.	85
4.3.1	Diagramas de casos de uso.	85

4.3.1.1	Diagramas de caso de uso – Usuario impedido.	86
4.3.1.2	Diagramas de caso de uso – Usuario no impedido.	86
4.3.2	Descripción textual de los casos de uso.	86
4.3.2.1	Descripción de caso de uso – Usuario impedido.....	88
4.3.2.2	Descripción de caso de uso – Usuario no impedido (Extra).....	89
5.	Descripción del sistema Domótico.	91
5.1	Introducción.	91
5.2	Requisitos del sistema domótico.	92
5.3	Infraestructura del sistema.	94
5.3.1	Servidor domótico.....	95
5.3.2	Fuente de alimentación (Bus).	95
5.3.3	Actuador.....	95
5.3.4	Sensor.....	96
5.3.5	Módulo de comunicación USB.	96
5.3.6	Otros componentes necesarios.....	97
6.	Diseño del Sistema.	99
6.1	Introducción.	99
6.2	Opciones de Diseño.....	99
6.2.1	Lenguajes de Programación.	99
6.2.2	Lenguajes de Bases de Datos.	101
6.2.3	Interfaz y Servicios.	101
6.3	Arquitectura del Sistema.....	105
6.4	Estructura de la Base de Datos.....	106
6.5	Diseño de UML del sistema.	107
6.6	Interfaces necesarios para el sistema.	108
7.	Implementación del Sistema.....	111
7.1	Introducción.	111
7.2	Herramientas de desarrollo del sistema.	111
7.2.1	Eclipse.....	111
7.2.2	SDK Android.....	112
7.2.3	SQL Admin.	112
7.2.4	Microsoft Office 2010.....	113
7.3	Servicios necesarios para el sistema.	113

7.3.1	Google Voice Recognizer:.....	113
7.3.2	Text To Speech:	114
7.3.3	Google Translate ^[6] :	114
7.3.4	HTTP Request ^[7] :	115
7.4	Sistema Android.	117
7.4.1	Clases Java del sistema.....	117
7.4.2	Base de Datos en SQLite.....	118
8.	Pruebas del Sistema.	119
8.1	Introducción.	119
8.2	Procedimientos de pruebas.	119
8.2.1	Pruebas de infraestructura hardware.	120
8.2.2	Pruebas de flujo de la aplicación.....	122
8.2.3	Pruebas de funcionalidad.....	125
9.	Conclusiones y Líneas Futuras.....	129
9.1	Conclusiones.....	129
9.2	Líneas futuras de mejora.....	130
Anexo I – Manual de Usuario.		131
Introducción.		131
Modo de Uso.....		131
Modo automático.....		132
Modo manual.		135
Anexo II – Manual de Instalación.		137
Introducción.		137
Pasos para la instalación.		137
Market de Android.		138
Categoría y Disponibilidad.....		139
Descarga e Instalación.....		140
Anexo III – Acrónimos, definiciones y referencias.		141
Acrónimos y definiciones.		141
Acrónimos.		141
Definiciones.....		142
Referencias.....		144

Índice de Tablas:

Tabla 1.: Cuota del mercado según las versiones del sistema operativo Android.....	46
Tabla 2.: Presupuesto de los recursos humanos.....	63
Tabla 3.: Presupuesto de los materiales.	64
Tabla 4.: Presupuesto del software necesario.	64
Tabla 5.: Presupuesto del software necesario.	64
Tabla 6.: Presupuesto del producto.	65
Tabla 7.: Presupuesto, formas de Pago.....	65
Tabla 8.: Plantilla de Requisito de Usuario.....	69
Tabla 9.: Requisito de Usuario de Capacidad 01.....	69
Tabla 10.: Requisito de Usuario de Capacidad 02.....	69
Tabla 12.: Requisito de Usuario de Capacidad 04.....	69
Tabla 13.: Requisito de Usuario de Capacidad 05.....	70
Tabla 14.: Requisito de Usuario de Capacidad 06.....	70
Tabla 15.: Requisito de Usuario de Capacidad 07.....	70
Tabla 16.: Requisito de Usuario de Capacidad 08.....	70
Tabla 17.: Requisito de Usuario de Capacidad 09.....	70
Tabla 18.: Requisito de Usuario de Capacidad 11.....	70
Tabla 19.: Requisito de Usuario de Capacidad 11.....	71
Tabla 20.: Requisito de Usuario de Capacidad 12.....	71
Tabla 21.: Requisito de Usuario de Capacidad 13.....	71
Tabla 22.: Requisito de Usuario de Capacidad 14.....	71
Tabla 23.: Requisito de Usuario de Capacidad 15.....	71
Tabla 24.: Requisito de Usuario de Capacidad 16.....	71
Tabla 25.: Requisito de Usuario de Capacidad 17.....	72
Tabla 26.: Requisito de Usuario de Restricción 01.....	72
Tabla 27.: Requisito de Usuario de Restricción 02.....	72
Tabla 28.: Requisito de Usuario de Restricción 03.....	72
Tabla 29.: Requisito de Software funcional 01.	73
Tabla 30.: Requisito de Software funcional 02.	74
Tabla 31.: Requisito de Software funcional 03.	74
Tabla 32.: Requisito de Software funcional 04.	74
Tabla 33.: Requisito de Software funcional 05.	74
Tabla 34.: Requisito de Software funcional 06.	74
Tabla 35.: Requisito de Software funcional 07.	75
Tabla 37.: Requisito de Software funcional 09.	75
Tabla 38.: Requisito de Software funcional 10.	75
Tabla 39.: Requisito de Software funcional 11.	75
Tabla 40.: Requisito de Software funcional 12.	76
Tabla 41.: Requisito de Software funcional 13.	76
Tabla 42.: Requisito de Software funcional 14.	76
Tabla 43.: Requisito de Software funcional 15.	76

Tabla 44.: Requisito de Software funcional 16.	76
Tabla 45.: Requisito de Software funcional 17.	77
Tabla 46.: Requisito de Software funcional 18.	77
Tabla 47.: Requisito de Software funcional 19.	77
Tabla 48.: Requisito de Software funcional 20.	77
Tabla 49.: Requisito de Software funcional 21.	77
Tabla 50.: Requisito de Software funcional 22.	78
Tabla 51.: Requisito de Software funcional 23.	78
Tabla 52.: Requisito de Software funcional 24.	78
Tabla 53.: Requisito de Software funcional 25.	78
Tabla 54.: Requisito de Software funcional 26.	78
Tabla 55.: Requisito de Software funcional 27.	79
Tabla 56.: Requisito de Software funcional 28.	79
Tabla 57.: Requisito de Software funcional 29.	79
Tabla 58.: Requisito de Software funcional 30.	79
Tabla 59.: Requisito de Software funcional 31.	79
Tabla 60.: Requisito de Software funcional 32.	80
Tabla 61.: Requisito de Software funcional 33.	80
Tabla 62.: Requisito de Software funcional 34.	80
Tabla 63.: Requisito de Software funcional 35.	80
Tabla 64.: Requisito de Software funcional 36.	80
Tabla 65.: Requisito de Software funcional 37.	81
Tabla 66.: Requisito de Software funcional 38.	81
Tabla 67.: Requisito de Software funcional 39.	81
Tabla 68.: Requisito de Software funcional 40.	81
Tabla 69.: Requisito de Software no funcional 01.	81
Tabla 70.: Requisito de Software no funcional 02.	82
Tabla 71.: Requisito de Software no funcional 03.	82
Tabla 72.: Requisito de Software no funcional 04.	82
Tabla 73.: Requisito de Software no funcional 05.	82
Tabla 74.: Requisito de Software no funcional 06.	82
Tabla 75.: Matriz de trazabilidad de requisitos de usuario de capacidad y requisitos de software funcionales.	83
Tabla 76.: Matriz de trazabilidad de requisitos de usuario de restricción y requisitos no funcionales.	84
Tabla 77.: Plantilla de descripción de los casos de uso.	87
Tabla 78.: CU-1.0 Mandato de elemento por voz.	88
Tabla 79.: CU-1.1 Mandato de elemento por voz.	89
Tabla 80.: CU-2.0 Mandato de elemento por selección táctil.	89
Tabla 81.: CU-2.1 Mandato de elemento por selección táctil.	90
Tabla 82.: Comparación entre HTML5 y Java.	100
Tabla 83.: Comparación de servicios de traducción.	103
Tabla 84.: Plantilla de Pruebas del sistema.	120
Tabla 85.: Prueba PHAR – 01.	120

Tabla 86.: Prueba PHAR – 02.....	121
Tabla 87.: Prueba PHAR – 03.....	121
Tabla 88.: Prueba PHAR – 04.....	121
Tabla 89.: Prueba PFLU – 01.....	122
Tabla 90.: Prueba PFLU – 02.....	122
Tabla 91.: Prueba PFLU – 03.....	122
Tabla 92.: Prueba PFLU – 04.....	123
Tabla 93.: Prueba PFLU – 05.....	123
Tabla 94.: Prueba PFLU – 06.....	123
Tabla 95.: Prueba PFLU – 07.....	124
Tabla 96.: Prueba PFLU – 08.....	124
Tabla 97.: Prueba PFUN – 01.....	125
Tabla 98.: Prueba PFUN – 02.....	125
Tabla 99.: Prueba PFUN – 03.....	126
Tabla 100.: Prueba PFUN – 04.....	126
Tabla 101.: Prueba PFUN – 05.....	127

Índice de Ilustraciones:

<i>Ilustración 1: Ejemplo de casa domótica.</i>	26
<i>Ilustración 2: Número de activaciones de dispositivos Android según el año.</i>	39
<i>Ilustración 3: Robot “Andy”, logotipo de ‘Android’.</i>	40
<i>Ilustración 4: Cuota de mercado según la versión del sistema operativo.</i>	45
<i>Ilustración 5: Ilustración de la aplicación de Securitas Direct.</i>	49
<i>Ilustración 6: Ilustración de la aplicación del fabricante Crestron.</i>	49
<i>Ilustración 7: Ilustración de la aplicación del fabricante ThinkFlood.</i>	51
<i>Ilustración 8: Diagrama del proyecto de domótica asistencial de ‘ITADA’.</i>	52
<i>Ilustración 9: Métrica v3 ilustración proporcionada por el Ministerio de España ^[2].</i>	54
<i>Ilustración 10: Ciclo de vida del proyecto.</i>	59
<i>Ilustración 11: Organigrama del equipo de trabajo.</i>	59
<i>Ilustración 12: Planificación de la primera iteración del proyecto.</i>	61
<i>Ilustración 13: Planificación de la segunda iteración del proyecto.</i>	61
<i>Ilustración 14: Planificación de la tercera iteración del proyecto.</i>	61
<i>Ilustración 15: Diagrama de casos de uso de un usuario impedido.</i>	86
<i>Ilustración 16: Diagrama de casos de uso de un usuario no impedido.</i>	86
<i>Ilustración 17: Esquema que resume las funciones que se interconectarán.</i>	92
<i>Ilustración 18: Esquema del sistema domótico asistencial.</i>	94
<i>Ilustración 19: Central IP de la instalación.</i>	95
<i>Ilustración 20: Fuente de Alimentación de la instalación.</i>	95
<i>Ilustración 21: Actuador de la instalación.</i>	95
<i>Ilustración 22: Actuador de la instalación.</i>	96
<i>Ilustración 23: Módulo de comunicación USB de la instalación.</i>	96
<i>Ilustración 24: Otros componentes de la instalación.</i>	97
<i>Ilustración 25: Primera interfaz del sistema.</i>	101
<i>Ilustración 26: Segunda interfaz del sistema.</i>	102
<i>Ilustración 27: Tercera interfaz del sistema.</i>	102
<i>Ilustración 28: Arquitectura del sistema.</i>	105
<i>Ilustración 29: Estructura de base de datos del sistema.</i>	106
<i>Ilustración 30: Sistema UML del sistema.</i>	107
<i>Ilustración 31: Interfaces de usuario y su flujo del modo automático.</i>	108
<i>Ilustración 32: Interfaces de usuario y su flujo del modo manual.</i>	109
<i>Ilustración 33: Uso de Google Voice Recognizer.</i>	113
<i>Ilustración 34: Uso de Google Text To Speech.</i>	114
<i>Ilustración 35: Logo de Google Translate.</i>	114
<i>Ilustración 36: Logo de Java, API utilizada para el servicio HTTP Request.</i>	115
<i>Ilustración 37: Diagrama de uso de servicios.</i>	116
<i>Ilustración 38: Sistema de ficheros de la aplicación.</i>	117
<i>Ilustración 39: Icono de la aplicación a ejecutar.</i>	131
<i>Ilustración 40: Captura de pantalla de inicio de la aplicación.</i>	132
<i>Ilustración 41: Captura de pantalla de la selección del modo automático de aplicación.</i>	133

<i>Ilustración 42: Captura de pantalla de la selección de acción en modo automático de aplicación.</i>	134
<i>Ilustración 43: Captura de pantalla de la selección y uso del modo manual de la aplicación. .</i>	135
<i>Ilustración 44: Captura de pantalla del icono de 'Play Store'.</i>	138
<i>Ilustración 45: Captura de pantalla del botón 'Aplicaciones'.</i>	139
<i>Ilustración 46: Captura de pantalla del menú 'Categoría'.</i>	139
<i>Ilustración 47: Captura de pantalla del botón 'Instalar'.</i>	140
<i>Ilustración 48: Captura de pantalla del botón 'Aceptar y Descargar'.</i>	140

1. Introducción.

En esta primera sección del documento, se pretenden describir las motivaciones que llevaron al equipo de desarrollo a afrontar este proyecto de trabajo de fin de grado. Además, se expondrán ciertas motivaciones que llevaron al autor del proyecto su elección como Trabajo de Fin de Grado (TFG de aquí en adelante).

No obstante, además de todo ello, se expondrán de forma breve y razonada los objetivos del sistema a desarrollar y de éste documento. De este modo, se podrá contextualizar al lector en la situación de la actualidad en la materia de domótica asistencial. Finalmente, se expondrá de forma breve la disposición del documento y su contenido en cada apartado.

1.1 Motivación.

En este apartado, se tratará de hacer una síntesis de dos cuestiones importantes. La primera, la motivación del proyecto a nivel general, seguidamente, se hará de la motivación a nivel personal que llevó a la elección de este proyecto.

Pues bien, a nivel general, el proyecto resulta muy interesante desde el punto de vista domótico. Se encontraron una serie de necesidades para un sistema domótico y se trataron de dar solución por medio de la integración de diversas soluciones.

El sistema final deseado, deberá contar con un módulo visual, otro auditivo y finalmente un sistema domótico con la infraestructura necesaria para la automatización de las acciones de los elementos. Como se puede observar, la integración, es muy completa.

Desarrollar una aplicación móvil para el sistema operativo Android enlazado con el sistema domótico, es una gran motivación para el equipo de implicado en general. Además, el sistema puede observarse como un nexo de unión entre la tecnología y las personas discapacitadas, por lo que la labor social también, es considerable.

Hoy en día, el número de aplicaciones móviles crece a un nivel vertiginoso, y en los últimos años, se han creado diversas aplicaciones para terminales móviles con la capacidad de realizar tareas para su usuario. De este modo, se implementan

distintas aplicaciones accesibles a través de un terminal móvil tipo Smartphone. Pues bien, hasta donde alcanza mi conocimiento, no hay ninguna aplicación desarrollada para este fin. Esta, puede ser una buena opción para decidir desarrollar éste sistema.

A nivel personal, la motivación que me ha llevado a desarrollar este proyecto, se basa principalmente, en el hecho de querer formar parte de un proyecto social que ayude a mejorar la calidad de vida de una persona con discapacidad. A partir de ahí, existen multitud de motivaciones tanto personales como sociales que pueden ser aportadas.

Pero la razón por la cual no dudé ni un momento en aceptar el proyecto, mi tía abuela M^a Ángeles, símbolo de la lucha para mí, ella, es la verdadera culpable. Su infancia la pasó de ambulatorio en hospital, y viceversa. Debido a una serie de caídas que sufrió su tibia y peroné se fracturaron en múltiples posiciones y ocasiones. Hoy en día con 86 años de los cuales unos 70 de ellos viviendo sola, sé que no podrá disfrutar del sistema creado. No obstante, el hecho de ayudar a personas y familias que se puedan encontrar en nuestra misma situación, ya es motivo de satisfacción personal.

Además, de ésta, se podrían enumerar otras, desde luego no tan importantes para mí a nivel personal pero con un gran peso social, así como motivaciones legales que deberían de llevar a la creación de este tipo de sistemas.

La ley de Abril del 2009 de ayuda a la dependencia, establece en nuestro territorio una serie de servicios y derechos que se han de prestar a las personas con discapacidad en nuestro país. A pesar de que éstas están siendo solicitadas por las personas que así lo requieren, este tipo de sistemas no son fomentados a pesar del gran potencial tecnológico del país. Por lo que puede suponer otra motivación más, la creación de este tipo de sistemas para las casas de las personas con algún tipo de dependencia.

1.2 Objetivos.

El principal objetivo del proyecto, es el del diseño y la integración en el sistema operativo Android, de un módulo de reconocimiento de voz para un sistema de domótica asistencial

Se puede entender por tanto, que otro de los objetivos que tiene este proyecto es la de desarrollar un buen módulo de reconocimiento de voz para que éste, pueda ser utilizado en un sistema domótico real.

Otro de los objetivos del proyecto, se puede entender como la integración de dicho módulo de reconocimiento de voz con otro módulo de reconocimiento visual.

Finalmente, el proyecto no pierde de vista, el objetivo principal de la aplicación, que es el de fomentar la autosuficiencia de un discapacitado en un entorno domótico. De éste modo, el objetivo final del proyecto es, que una persona discapacitada pueda controlar mediante un módulo de reconocimiento por voz todo el entorno domótico que lo rodea en su hogar.

1.3 Contexto socio-cultural y económico.

En la actualidad, los sistemas domóticos cada vez ofrecen más posibilidades a sus usuarios. Hoy día, no resulta extraño que en viviendas de nueva construcción, venga pre-instalado o instalado un módulo de domótica para la vivienda. Esto significa que el contexto actual, invita a que empresas tecnológicas empiecen a dar solución a estas viviendas con nuevas funcionalidades.

De este modo, las ideas para mejorar sistemas domóticos han ido en crecimiento junto con el crecimiento de las tecnologías. Hoy en día existen multitud de dispositivos móviles con el sistema operativo Android pre-instalado. Por lo que se puede afirmar que el desarrollo de ésta aplicación es un aporte más a una gran gama de posibilidades que abren este tipo de dispositivos.

Por otro lado, el contexto de los sistemas de ayudas a los discapacitados en nuestro país, es cuanto menos, curioso. En España, se estrenó una ley de la dependencia a discapacitados^[1], en la que se sostiene que se debe dar accesibilidad a todos los medios a las personas discapacitadas según su grado. Para ello, los servicios más demandados son los de asistencia médica inmediata o la de

personas que han de llevar a cabo todas las labores del hogar. Decir, que las ayudas en este sentido, no han cubierto las expectativas con las que fueron creadas y muchas personas con movilidad reducida, nunca han podido disfrutar de las mismas. Por ello, este proyecto podría ayudar a mejorar muchos de éstos aspectos e inclusive las instituciones públicas podrán beneficiarse de ello.

Finalmente, decir que, hasta donde se ha podido comprobar, no existe ningún sistema domótico en el que se conjunten los sistemas de una ‘casa inteligente’ y los de las aplicaciones móviles para personas con discapacidad. De este modo, el contexto socio-cultural y económico en el que ha surgido este proyecto es desde la investigación y el desarrollo, con los gastos que ello conlleva.

1.4 Estructura del documento.

En este apartado, se pretende dar una breve pero profunda descripción del contenido de cada uno de los puntos que se exponen en esta memoria. Con este apartado, se pretende que el lector tenga una idea de cada uno de los puntos con anterioridad a la lectura de los mismos.

El apartado dos de esta memoria, es “Estado del arte”. Éste, es un apartado dedicado a situar al lector en el contexto actual. Es decir, se trata de describir de forma breve la situación actual de las tecnologías/sistemas necesarios para la implementación de este proyecto.

Seguidamente, surge el apartado número tres, el cual es “Gestión del proyecto”, en él, se tratarán aspectos relativos a la gestión de tiempos de análisis, diseño, implementación, así como los de pruebas.

En el cuarto apartado, “Análisis del Sistema”, se realizará un análisis en profundidad del sistema en el cual se especificarán los requisitos que se han extraído del sistema.

En el siguiente apartado, el quinto, “Descripción del sistema domótico”, será el encargado de realizar una breve explicación del sistema domótico en su conjunto y ciertos requisitos que éste, ha de cumplir.

El sexto de los apartados, es “Diseño del sistema”, en este punto, se dará solución a todos los aspectos de diseño del propio sistema. De esta forma,

quedarán expuestos antes de abarcar los apartados de implementación y pruebas del sistema.

A continuación, se exponen dos apartados, claves en la memoria antes de abarcar unas conclusiones. El primero, el de “Implementación del sistema”, en el cual, se expondrán las fases de implementación y demás detalles relacionados. El segundo, el de “Pruebas del sistema”, uno de los más importantes, un apartado, en el que se expondrán los tipos de pruebas, las pruebas en sí y el resultado de las mismas.

Para completar los puntos enumerados, llegamos al noveno, “Conclusiones y líneas futuras”. En él, se realizará un brece resumen de las conclusiones sacadas durante el desarrollo del proyecto así como las futuras líneas que se han de seguir para continuar evolucionando el producto.

Finalmente, se exponen una serie de anexos, los cuales contienen contenidos esenciales para el documento. En primer lugar, el “Anexo I” el cual, contendrá el manual de usuario, así como el “Anexo II” encargado del manual de instalación. Finalmente quedará, el “Anexo III”, que contendrá una serie de definiciones acrónimos y referencias

2. Estado del arte.

En este apartado, se contextualizará al lector en el panorama actual de los teléfonos inteligentes, así como una pequeña introducción a los sistemas de domótica de la actualidad.

Para ello, se comenzará por una introducción a los sistemas domóticos, sus infraestructuras y tipos de arquitecturas, se continuará con la ley actual de ayuda a la dependencia^[1], y finalmente, se concluirá con una introducción a los sistemas móviles, más concretamente al sistema operativo Android.

2.1 Sistemas de Domótica

2.1.1 Introducción

Se entiende como domótica el conjunto de sistemas capaces de automatizar una vivienda. De este modo, se le aporta servicios de gestión energética, seguridad, bienestar y comunicación que pueden ser integrados por medio de redes interiores y exteriores de comunicación. Estas comunicaciones pueden ser inalámbricas o cableadas, cuyo control goza de cierta ubicuidad desde dentro y fuera del hogar. De forma sencilla, se podría definir como:

“La integración de la tecnología en el diseño inteligente de un recinto cerrado”

El término, viene de la unión de dos palabras como ‘domus’ (que quiere decir, casa, en latín) y tica (de automática, palabra que en latín quiere decir ‘que funciona por sí sola’). En España la domótica tiene presencia mediante multitud de empresas, algunas con más de 12 años en el mercado.



Ilustración 1: Ejemplo de casa domótica.

2.1.2 Servicios en sistemas de domótica

Principalmente, los servicios que ofrecen este tipo de sistemas se pueden clasificar en cinco ámbitos:

2.1.2.1 Ahorro energético

El ahorro energético no es algo tangible, sino un concepto al que se puede llegar de muchas maneras. En muchos casos no es necesario sustituir los aparatos o sistemas del hogar por otros que consuman menos sino una gestión eficiente de los mismos mediante distintos mecanismos:

- Climatización: Programación y zonificación.
- Gestión eléctrica:
 - Racionalización de cargas eléctricas: Desconexión de equipos de uso no prioritario en función de cierto consumo eléctrico en cierto tiempo.
 - Gestión de tarifas: Derivando el funcionamiento de los aparatos de mayor consumo energético a ciertas horas del día en el cual la tarifa es más barata.
- Uso de energías renovables.

2.1.2.2 Confort:

El confort, conlleva todas las actuaciones que se puedan llevar a cabo que mejoren las condiciones de comodidad de una vivienda. Dichas actuaciones, pueden ser de carácter activo, pasivo o mixto:

- Iluminación:
 - Apagado general de todas las luces de la vivienda.
 - Automatización del apagado o encendido en cada foco de luz.
 - Regulación de la iluminación según necesidades ambiente.
- Automatización: Los distintos sistemas/instalaciones/equipos dotándolos de un control eficiente y de fácil manejo.
- Integración del portero al teléfono o del video portero al televisor.
- Control vía internet.
- Gestión Multimedia del ocio electrónico.
- Generación de programas de forma sencilla para el usuario.

2.1.2.3 Seguridad

Consiste en una red de seguridad encargada de proteger tanto los bienes patrimoniales como la seguridad personal.

- Alarmas de intrusión (Anti-intrusión): Se utilizan para detectar o prevenir la presencia de personas extrañas en una vivienda o edificio.
 - Detección de un posible intruso (Detectores volumétricos o perimetrales)
 - Cierre de persianas puntual y seguro
 - Simulación de presencia
- Alarmas de detección de incendios, fugas de gas, escapes de agua, concentración de monóxido en garajes cuando se usan vehículos de combustión.
- Alerta médica, tele asistencia.
- Acceso a cámaras IP.

2.1.2.4 Comunicaciones

Los sistemas o infraestructuras de comunicaciones que posee el hogar.

- Ubicuidad en el control tanto externo como interno, control remoto desde Internet, PC, mandos inalámbricos (p.ej. PDA con WiFi), aparellaje eléctrico.
- Tele asistencia
- Tele mantenimiento
- Informes de consumo y costes
- Transmisión de alarmas.
- Intercomunicaciones.

2.1.2.5 Accesibilidad

Bajo este epígrafe se incluyen las aplicaciones o instalaciones de control remoto del entorno que favorecen la autonomía personal de personas con limitaciones funcionales, o discapacidad.

El concepto "diseño" para todos es un movimiento que pretende crear la sensibilidad necesaria para que al diseñar un producto o servicio se tengan en cuenta las necesidades de todos los posibles usuarios, incluyendo las personas con diferentes capacidades o discapacidades, es decir, favorecer un diseño accesible para la diversidad humana.

La inclusión social y la igualdad son términos o conceptos más generalistas y filosóficos. La domótica aplicada a favorecer la accesibilidad es un reto ético y creativo pero sobre todo es la aplicación de la tecnología en el campo más necesario, para suplir limitaciones funcionales de las personas.

El objetivo no es que las personas con discapacidad puedan acceder a estas tecnologías, porque las tecnologías en si no son un objetivo, sino un medio. El objetivo de estas tecnologías es favorecer la autonomía personal. Los destinatarios de estas tecnologías son todas las personas, ya que por enfermedad o envejecimiento, todos somos o seremos discapacitados, más pronto o más tarde.

2.1.3 El sistema domótico

2.1.3.1 Dispositivos

Los dispositivos que componen un sistema de domótica tradicional, incluyen principalmente tres tipos:

- Controladores.
- Sensores.
- Actuadores.

2.1.3.2 Arquitectura

Desde el punto de vista donde reside la inteligencia del sistema de domótica, hay varias arquitecturas distintas.

- **Arquitectura Centralizada:** un controlador centralizado recibe información de múltiples sensores y, una vez procesada, genera las órdenes oportunas para los actuadores.
- **Arquitectura Distribuida:** toda la inteligencia del sistema está distribuida por todos los módulos sean sensores o actuadores. Suele ser típico de los sistemas de cableado en bus, o redes inalámbricas.
- **Arquitectura mixta:** sistemas con arquitectura descentralizada en cuanto a que disponen de varios pequeños dispositivos capaces de adquirir y procesar la información de múltiples sensores y transmitirlos al resto de dispositivos distribuidos por la vivienda, p.ej. aquellos sistemas basados en Zigbee y totalmente inalámbricos.

2.1.3.3 Elementos de una instalación domótica

- Central de gestión
- Sensores
- Actuadores
- Soportes de comunicación
- Aparatos terminales

2.1.4 Asociaciones Internacionales y Españolas de domótica

CENELEC:

La Comisión CENELEC/ENTR/e-Europe/2001-03 es la encargada de elaborar normas a nivel internacional y la organización que ha promocionado el Smart House Forum.

CEDOM:

Asociación Española de Domótica. Su objetivo principal es la promoción de la Domótica. Se trata del foro nacional en el que se reúnen todos los agentes del sector en España: fabricantes de productos domóticos, fabricantes de sistemas, instaladores, integradores, arquitecturas e ingenierías, centros de formación, universidades, centros tecnológicos.

LonUsers España

Asociación de usuarios de la tecnología LonWorks, siendo creada por la iniciativa de empresas líderes en los diferentes sectores de aplicación de la tecnología LonWorks (domótica, inmótica, control industrial y de transporte).

AENOR

Asociación que ha creado también una Subcomisión del Hogar Digital, dentro de la Comisión 133 (AEN/CTN 133 Telecomunicaciones) a fin de definir estándares.

2.1.5 Hogar Digital

El hogar digital es una vivienda que a través de equipos y sistemas, y la integración tecnológica entre ellos, gracias a la domótica, ofrece a sus habitantes funciones y servicios que facilitan la gestión y el mantenimiento del hogar, aumentan la seguridad; incrementan el confort; mejoran las telecomunicaciones; ahorran energía, costes y tiempo, y ofrecen nuevas formas de entretenimiento, ocio y otros servicios dentro de la misma y su entorno sin afectar a las casas normales.

2.1.5.1 Diferencias entre casa domótica y hogar digital

Una vivienda domótica dispone de un gran número de equipos y sistemas, principalmente autónomos, a los que hay que sumar diferentes redes, como la telefonía, las redes de datos (cableadas e inalámbricas), la televisión, electrodomésticos, equipamiento de audio y video, calefacción, aire-condicionado, seguridad, riego, iluminación, etc.

Para convertirse en un hogar digital, a una casa domótica le faltaría la convergencia de las comunicaciones, la informática y el entretenimiento gracias a las redes de banda ancha es una tendencia consolidada a nivel mundial.

Es necesario resaltar que el Hogar Digital no consiste simplemente en la instalación de dispositivos para controlar determinadas funciones de las viviendas, tales como alarmas, iluminación, climatización, etc.

2.2 Ley de dependencia (20 de Abril del 2006) ^[1].

Esta ley, se ha creado como una nueva modalidad de protección social, que amplía y complementa la acción protectora del Estado y del Sistema de la Seguridad Social ya existentes.

De este modo, se pretende atender a zonas dependientes, ya sea por su edad (envejecimiento del sujeto) o por razones de enfermedad, discapacidad o limitación en otros. Para ello se ha creado el Sistema para la Autonomía y Atención a la Dependencia (SAAD), que debe garantizar las condiciones básicas y la previsión de los niveles de protección referidos en esta ley.

El objeto de esta ley es regular las condiciones básicas para garantizar la igualdad de los ciudadanos en el derecho de la promoción de la autonomía personal y atención a dependientes.

Se crea también un Consejo Territorial del SAAD que acordará el marco de cooperación entre las instituciones públicas, la intensidad de los servicios del catálogo, condiciones y cuantías de las prestaciones económicas, criterios de participación de los beneficiarios y el baremo para el reconocimiento de la situación de dependencia.

2.2.1 Prestaciones y catálogo de servicios para personas dependientes ^[1].

Mediante esta ley, se pretende otorgar a las personas que cumplen una serie de requisitos las prestaciones y servicios necesarios para poder desarrollar una vida con normalidad. Es por ello, que los servicios han de ser variados y completos, la ley establece el siguiente catálogo de servicios:

1. Los Servicios de Prevención
2. Los Servicio de Teleasistencia
3. Servicio de Ayuda a Domicilio
 - Atención de las necesidades del hogar.
 - Cuidados personales.
4. Servicio de Centro de Día y de Noche
 - Centro de Día para mayores.
 - Centro de Día para menores de 65 años.
 - Centro de Día de atención especializada.
 - Centro de Noche.
5. Servicio de Atención Residencial

- Residencia de personas mayores en situación de dependencia.
- Centro de atención a personas en situación de dependencia, en razón de los distintos tipos de discapacidad.

2.2.2 Baremos de aplicación de la ley ^[1].

La puntuación final de este baremo permitirá incluirse en alguno de los tres grados de dependencia:

1. Grado I:
 - Nivel 1: 25-39 puntos
 - Nivel 2: 40-49 puntos
2. Grado II:
 - Nivel 1: 50-64 puntos
 - Nivel 2: 65-74 puntos
3. Grado III:
 - Nivel 1: 75-89 puntos
 - Nivel 2: 90-100 puntos

Se puede aplicar en cualquier situación de discapacidad a partir de los 3 años de edad.

2.2.3 Aplicabilidad real de la ley y situación actual.

El estado español, aportará 12.640 millones de euros desde 2007 hasta 2015. El primer año, serán unos 400 millones, siendo esta cantidad aumentada hasta sumar 2.212 millones más en 2.015.

Las comunidades autónomas deberán aportar las mismas cantidades. Por tanto, los usuarios sufragarán el 35% del total en función de su renta personal. En caso de que una familia pudiese pagar el 90% del coste de una residencia así deberá hacerlo durante el máximo tiempo posible.

Además, los recortes en el presupuesto general del estado que se están produciendo este año 2012, están mermando aún más el presupuesto para estas prestaciones.

Quizás por estas y por alguna razón más, la ley, no está funcionando como se esperaba.

2.3 Sistemas Móviles

Los sistemas móviles en la actualidad, gozan de gran diversidad de dispositivos en los que ser implantados. Se pueden ver dispositivos móviles en diversas formas. Se inició su desarrollo mediante pequeñas video consolas móviles como la "super nintendo" o la "PSP" de Sony Entertainment.

Continuó con el desarrollo de reproductores musicales cada vez más inteligente, que acabaron reproduciendo aplicaciones, videos, y son capaces de acceder a internet. Finalmente, hoy en día, se pueden encontrar sistemas móviles en forma de teléfonos móviles o simplemente de tabletas electrónicas.

2.3.1 Introducción

Un sistema operativo (SSOO) de un dispositivo móvil, es el programa base con el cual dicho dispositivo, funciona. Si los comparamos con las computadoras, estaríamos hablando de Windows, Mac o incluso Linux. Actualmente, somos testigos de una nueva guerra digital y las empresas, pelean por conseguir ser parte de tu teléfono móvil, y qué mejor forma de hacerlo que desde la base, el sistema operativo.

Con la creciente aparición de teléfonos inteligentes, y dispositivos móviles en general cada vez más pequeños, están más cerca de una mini computadora portátil que de un teléfono o un reproductor de contenido multimedia. Es por ello, que cada vez se necesitará de nuevos Sistemas Operativos avanzados y con mayor flexibilidad en el mercado.

Con esa mentalidad y debido a la importancia de éstos sistemas, las empresas, se iniciaron en el desarrollo de este tipo de Sistemas Operativos y parece que la tendencia fue por los Sistemas Operativos con Código Abierto (Open Source Operating Systems). Lo cual, no significa que sean gratuitos, sino que su código estará abierto a la vista de todos. De éste modo, se consigue que cualquiera pueda desarrollar nuevas aplicaciones o pulgins.

La filosofía del consumidor contribuidor no es nueva, de hecho lo que se hizo fue copiar el proceder de ciertos sistemas operativos convencionales que tienen dicha filosofía. De este modo, cualquier desarrollador con una mínima experiencia, se pretende, puede ayudar y apoyar al sistema operativo mediante nuevas aplicaciones o accesorios que mejoren dicho sistema operativo.

2.3.2 Dispositivos Móviles

Hoy en día, se entiende por dispositivos móviles ciertos elementos con suficiente autonomía de cómputo, energética y que están conectados a una red. Existen más características para definirlos, como:

- Son aparatos pequeños.
- Con capacidades de procesamiento.
- Con conexión permanente o intermitente a una red.
- Con memoria limitada.
- Diseñados específicamente para una función, pero que pueden llevar a cabo otras más generales.
- Normalmente se asocian al uso individual de una persona, tanto en posesión como en operación, el cual puede adaptarlos a su gusto.
- La mayoría de estos aparatos pueden ser transportados en el bolsillo del propietario.

Una de las características más importantes de las anteriormente expuestas es que el dispositivo móvil puede ser transportado en el bolsillo de propietario (normalmente) y además suele ser con conexión permanente o intermitente a una red, como puede ser Internet. Pues bien, un dispositivo móvil inalámbrico, es aquel que es capaz de comunicarse o acceder a una red sin cables. Éste tipo de dispositivos se comportan como si estuvieran directamente conectados a una red por medio de un medio físico (cable). De este modo, el dispositivo es capaz de almacenar datos y procesarlos.

Dadas las anteriores características, se puede entender como dispositivo móvil una computadora tradicional. No obstante, existen una serie de diferencias claras con las de un computador tradicional, estas, son diferencias como:

- Funcionalidad limitada.
- No necesariamente extensible y actualizable.
- En pocos años el usuario deberá cambiarlo.
- Más barato (según el dispositivo).
- Menos complicado en su manejo.
- Fácil de aprender su operación.
- No se requieren usuarios expertos.

2.3.3 Aplicaciones Móviles

Otra evolución destacable de los Sistemas Móviles, son sus aplicaciones. En el momento en que surgió un mercado tan potente como el de los Smartphone o dispositivos móviles con gran capacidad de computación resultó evidente que se debía apostar por estas tecnologías. Tiempo después, se confirmó. Hoy en día más del 90% de los usuarios de dispositivos móviles inteligentes utilizan éste tipo de aplicaciones para el uso habitual de su dispositivo. De este modo, se demuestra que dichas aplicaciones tienen cabida en el mercado actual y futuro.

En un primer momento, éste tipo de aplicaciones, eran principalmente juegos recreativos que tenían una función puramente recreativa. Un juego, quizás algo que convierta la pantalla del teléfono en (por poner un ejemplo) el teclado de un piano o similares. Recientemente, una serie de aplicaciones útiles han despertado gran interés entre los usuarios. Entre ellas, figuran aplicaciones para el registro de gastos, manuales de diversas actividades y tareas y un sinfín de utilidades como convertidores de moneda. Finalmente, esta actitud ha desembocado en un sinfín de aplicaciones móviles que demuestran el potencial de los dispositivos en los que están instalados.

Para la gran parte de los casos, las aplicaciones móviles son creadas por diseñadores de otras empresas pero los comerciales, se encargan de la distribución, pago y comercialización limitada de la misma. Este acuerdo puede resultar muy ventajoso para ciertos comercios. El aumento del número de aplicaciones y sus funcionalidades llevan una relación directa en el aumento de computación y venta de los dispositivos móviles. Los diseñadores, también han encontrado nuevas formas de rentabilidad.

Por ejemplo, con la última versión de iPhone, se ha logrado vender contenido o servicios con recargo en las aplicaciones, atribuyendo a los comercios la tarea de cobrar tarifas únicas. Gracias al nuevo impulso que empieza a adquirir el mercado del móvil con las aplicaciones de marca, hay grandes perspectivas de obtener aún mayores ingresos.

A pesar de todas estas ventajas y el incremento en el uso de la telefonía móvil, algunos operadores de red móvil, temen la “canibalización” de los servicios existentes. Los operadores, vieron cómo decrementaron sus ingresos en servicios como los mensajes de texto (SMS, Short Message Service), o incluso las llamadas. Ello, se debió

principalmente a ciertas aplicaciones como el WhatsApp o el Viber. Por tanto, los principales inconvenientes a estas tecnologías según las operadoras, fueron:

- Aplicaciones del protocolo de transmisión de la voz por Internet (VoIP) que amenazan las llamadas telefónicas convencionales.
- Aplicaciones de mensajería instantánea que compiten con el servicio de mensajes breves (SMS).
- Aplicaciones que aceleran la navegación y reducen de esa forma el tráfico en línea.

Además, por miedo a los programas malintencionados de ciertos desarrolladores, las operadoras, dudan ante la posibilidad de dar acceso directo a las funcionalidades básicas del dispositivo a los programadores.

Finalmente, cabe apuntar que uno de los mayores escollos que se ha encontrado el desarrollo de aplicaciones móviles. Éste, ha sido la gran diversidad y ambigüedad de las restricciones impuestas para cada dispositivo, así como la falta de compatibilidad entre las distintas plataformas. Es decir, por ejemplo, productos elaborados para un dispositivo Android, han de ser re-elaborados para el sistema operativo iOS de Apple. Ello, se debe a las interfaces de programación o API's de las que hace uso.

2.3.4 Principales Sistemas Operativos Móviles

Los sistemas móviles en la actualidad, tienen una serie de ventajas frente a los sistemas operativos tradicionales que son innegables. No obstante, dentro de los sistemas operativos móviles, existe gran diversidad de opciones para cada dispositivo. Algún ejemplo de estos sistemas operativos, son, iOS, de Apple, Android, de Google, Windows Mobile Phone de Microsoft o WebOS incluso de HP.

Lo más esencial e importante en un sistema operativo de éstas características es el núcleo del mismo (kernel). Android, usa un kernel Linux con una mezcla especial de Java (máquina virtual de Java en Android, Dalvik). Los dispositivos móviles de Apple, por ejemplo, se basan en OS X, que a su vez, es una variante de Unix, uno de los sistemas operativos más poderosos del mundo de la informática. S60 y Windows Mobile son SSOO muy maduros y estables, a pesar de que la edad no siempre es una ventaja. Y por último, RIM usa un kernel propio que al igual que Android, contiene un

motor Java y a pesar de que han mejorado la interfaz notablemente a lo largo del tiempo, suele mostrar limitaciones.

A continuación, se van a exponer brevemente una serie de detalles básicos de los principales sistemas operativos:

- **iPhone OS / iOS (Apple):** El sistema operativo creado por la marca de Cupertino en primer lugar lo desarrolló para iPhone (iPhone OS). Hoy en día el sistema operativo compartido por la gran mayoría de sus dispositivos móviles es el iOS. Se trata de un SSOO desarrollado íntegramente por el equipo de Apple, y no está permitida su instalación en dispositivos de terceros.
- **Android OS (Google Inc.):** Éste sistema operativo para móviles, es el preferido por la gran mayoría de usuarios. Se trata de un sistema operativo de código libre y abierto. Existe una gran multitud de apps gratuitas o de bajo coste. Las actualizaciones para este sistema operativo son muy frecuentes. Implementa una máquina virtual de Java especial llamada 'Dalvik' la cual es la encargada de ejecutar todas las aplicaciones del dispositivo.
- **BlackBerry OS:** Es un sistema operativo móvil cerrado y desarrollado por la empresa "Research In Motion". Fue desarrollado para el uso exclusivo en terminales del fabricante 'BlackBerry'. Éste fabricante, está especializado en grandes corporaciones y ofrece a sus usuarios una gran capacidad de sincronización con Microsoft Exchange, Lotus Domino, Novell GroupWise y email.
- **MeeGo OS (Nokia and Intel):** Ambos fabricantes se unieron para obtener este sistema operativo de código libre y abierto. Está basado en dos tecnologías, la primera Maemo (Nokia) y la segunda Moblin (Intel), como se puede observar ninguno de los fabricantes quiso dejar de lado sus avances tecnológicos. Éste SSOO está diseñado para una cierta arquitectura de dispositivos, la ARMv7, usando microprocesadores Intel Atom.
- **Bada (Samsung Electronics):** Bada es un sistema operativo del cual es propietario Samsung que fue lanzado al mercado por primera vez en 2010. Samsung fue el primer fabricante en incluir en un dispositivo dicho SSOO. Principalmente Bada, es un SSOO que incluye novedades como gráficos 3D, gestos 'multi-touch'.

- **Palm OS (Garnet OS):** Se trata de un sistema operativo del cual Palm es propietario. Desarrollado originalmente en 1.996, las nuevas versiones de éste SSOO incluyen soporte para la expansión de puertos, nuevos procesadores y tarjetas de memoria externas. Además, son 100% compatibles con arquitecturas ARM y SmartPhones. Palm OS 5, fue actualizado para dar soporte a distintas resoluciones de pantalla, conexiones WiFi y nuevas capacidades multimedia.
- **Symbian OS (Nokia):** Symbian es un SSOO móvil muy ligado a los teléfonos que ofrecen una gran integración con comunicaciones y la administración información personal. Symbian, combina el 'middleware' con las comunicaciones inalámbricas con integración al buzón de entrada del correo electrónico. Nokia no mantiene Symbian como un proyecto de libre de código abierto.
- **Web OS (Palm/HP):** WebOS es un sistema operativo para móviles que corre una versión del kernel Linux. WebOS fue inicialmente desarrollado por Palm como sucesor de su sistema operativo PalmOS 5. Es un sistema operativo propietario de Mobile OS que fue adquirida por HP recientemente. Hoy en día, ofrece WebOS en dispositivos HP. Éste SSOO es el más utilizado en la gama de productos de HP.
- **Windows Mobile (Windows Phone 7):** Windows Mobile es un sistema operativo móvil cuyo propietario es Microsoft. Está usado en Smartphone y dispositivos móviles con o sin pantalla táctil. Basado en el kernel, Windows CE 5.2. En 2010 Microsoft anunció una nueva plataforma para dispositivos móviles Windows Phone 7.

Como se puede observar, todos los sistemas operativos móviles, son muy parecidos, no obstante, las características específicas de cada uno, les hacen más o menos atractivos a la hora de empezar a desarrollar aplicaciones sobre ellos. Finalmente, decir, que estudios de Junio de 2011, sitúan a Google como la mayor parte del mercado de sistemas móviles con un 40,1%, seguidamente la acompaña Apple con un 26,6% del mercado y finalmente RIM con un 23,4%. Los sistemas operativos de Microsoft (5,8%) y Nokia (Symbian 2%) concluyen la estadística. En nuestro caso, y dado que requerimos de una buena integración con Google, el sistema operativo que más nos pareció se amoldaba a nuestras pretensiones, fue Android.

Es por ello, que en el siguiente apartado realizaremos un análisis en profundidad de dicho SSOO, de cara a conocer más y más ampliamente dicho sistema operativo.

2.3.5 Sistema Operativo - Android^[3].

Se trata del sistema operativo basado en Linux que junto con aplicaciones middleware, está enfocado para ser utilizado en dispositivos móviles. Debido a su gran éxito, ha sido integrado en gran parte de los teléfonos inteligentes, tabletas, Google TV y diversos dispositivos más. Por lo general, este sistema, maneja aplicaciones como Google Play (Market de aplicaciones).

Desarrollado inicialmente por Android Inc., ésta empresa fue rápidamente comprada por Google en 2005. De este modo, se convirtió en el principal producto de la Open Handset Alliance, un conglomerado de fabricantes y desarrolladores de hardware, software y operadores de servicio. En la actualidad, el número de unidades vendidas de teléfonos inteligentes con Android, se ubican en el primer puesto de los EEUU y la gran parte de Europa. Mercados anteriormente copados por Apple y su teléfono inteligente iPhone. A nivel mundial alcanzó una cuota de mercado del 50,9% durante el cuarto trimestre de 2011, más del doble que el segundo sistema operativo (iOS de iPhone) con más cuota. Como se puede observar en el siguiente gráfico:



Ilustración 2: Número de activaciones de dispositivos Android según el año.

Hoy en día, la gran comunidad de desarrolladores con la que cuenta Android escribiendo aplicaciones hace extender de forma asombrosa la funcionalidad y potencial de éstos dispositivos. A la fecha, se han superado las 600.000 aplicaciones, teniendo en cuenta que dos tercios de éstas son gratuitas, únicamente contando las existentes en Google Play, si no sitios como Samsung Apps o el App Market. Con estos números en la mano, no es de extrañar que los usuarios prefieran este sistema operativo frente a sus competidores.

Programas escritos en el lenguaje de programación Java, no siguen un control exhaustivo de la seguridad de los mismos. Esto, produce la única debilidad de este sistema operativo, frente a sus competidores, el malware, no obstante, la mayor parte de éste es descargado en sitios de terceros.

2.3.5.1 Historia

Tanto el nombre Android (androide en su traducción al castellano) como Nexus One (primer teléfono inteligente de Google) hacen alusión a la novela de Philip K. Dick llamada “¿Sueñan los androides con ovejas eléctricas?”, adaptada al cine como, “Blade Runner”. Tanto es así, que el logotipo de Android, el robot “Andy”, es la forma de llamar en “Blade Runner” a los Androides.



Ilustración 3: Robot “Andy”, logotipo de ‘Android’.

Como ya se ha apuntado anteriormente, Google adquirió la empresa Android Inc., una pequeña empresa de El Palo, California en 2005. Tras ello, sus fundadores, se integraron al equipo de desarrollo de Google. Su adquisición, levantó sospechas y rumores que apuntaban que Google estaba planeando entrar en el mercado de los teléfonos inteligentes.

En ese momento, Google, mediante el equipo liderado por Rubin desarrolló una plataforma para dispositivos móviles basada en el kernel de Linux. Éste, fue promocionado a fabricantes de dispositivos y operadores con la promesa de proveer un sistema flexible y actualizable. En aquel momento, se verificó la cooperación entre fabricantes de hardware, software y las principales operadoras.

Finalmente, en Diciembre de 2006, los rumores de que Google entraría a formar parte del mercado de la telefonía móvil eran claros. La BBC y The Wall Street Journal señalaron que Google quería sus servicios de búsqueda y aplicaciones en los dispositivos móviles.

En septiembre de 2007, «InformationWeek» difundió un estudio de Evalueserve que reportaba que Google había solicitado diversas patentes en el área de la telefonía móvil.

Finalmente, el anuncio de Android como sistema operativo, se realizó el 5 de Noviembre de 2007 junto con la creación del consorcio ya comentado anteriormente, 'Open Handset Alliance' el cual, consta de 78 compañías de hardware, software y telecomunicaciones dedicadas al desarrollo de estándares abiertos para dispositivos móviles. Entre ellas, están compañías de lo más prestigiosas e históricas como, 'Texas Instruments', 'Broadcom Corporation', 'Nvidia', 'Qualcomm', o 'Samsung Electronics'.

Google como empresa liberó la gran parte del código de Android bajo la licencia Apache, la cual es libre y de código abierto, dejando así a los desarrolladores plena confianza.

2.3.5.2 Historial de Actualizaciones y Versiones^[4]

Como todo buen sistema operativo, Android, ha visto numerosas actualizaciones desde su liberación inicial. Estas actualizaciones al sistema operativo base típicamente arreglan defectos y agregan nuevas funciones. Generalmente cada actualización de Android es desarrollada bajo un nombre en código de un elemento relacionado con postres.

Los nombres en código están en orden alfabético:

1. Apple Pie
 - Versión 1.0
 - Liberado el 23 de septiembre de 2008.
2. Banana Bread
 - Versión 1.1
 - Liberado el 9 de Febrero de 2009.
3. Cupcake
 - Versión 1.5
 - Liberado el 30 de Abril del 2009.
 - Nuevas características para el usuario:
 - Posibilidad de grabar y reproducir videos a través del modo camcorder.
 - Capacidad de subir videos a YouTube e imágenes a Picasa directamente desde el teléfono.
 - Un nuevo teclado con predicción de texto
 - Soporte para Bluetooth A2DP y AVRCP
 - Capacidad de conexión automática para conectar a auricular Bluetooth a cierta distancia
 - Nuevos widgets y carpetas que se pueden colocar en las pantallas de inicio
 - Transiciones de pantalla animadas
4. Donut
 - Versión 1.6
 - Liberado el 15 de Septiembre del 2009.
 - De nuevo, una serie de actualizaciones:
 - Una experiencia mejorada en el Android Market.
 - Una interfaz integrada de cámara, filmadora y galería.
 - La galería ahora permite a los usuarios seleccionar varias fotos para eliminarlas.
 - Búsqueda por voz actualizada, con respuesta más rápida y mayor integración con aplicaciones nativas, incluyendo la posibilidad de marcar a contactos.
 - Experiencia de búsqueda mejorada que permite buscar marcadores, historiales, contactos y páginas web desde la pantalla de inicio.
 - Actualización de soporte para CDMA/EVDO, 802.1x VPN y text-to-speech.
 - Soporte para resoluciones de pantalla WVGA.
 - Mejoras de velocidad en las aplicaciones de búsqueda y cámara

5. Eclair

- Versión 2.0 y 2.1.
- Liberada el 26 de Octubre del 2009.
- Cambios producidos:
 - Velocidad de hardware optimizada.
 - Soporte para más tamaños de pantalla y resoluciones.
 - Interfaz de usuario renovada.
 - Nuevo interfaz de usuario en el navegador y soporte para HTML5.
 - Nuevas listas de contactos.
 - Una mejor relación de contraste para los fondos.
 - Mejoras en Google Maps 3.1.2.
 - Soporte para Microsoft Exchange.

6. Froyo

- Versión 2.2
- Liberación el 20 de Mayo del 2010.
- Nuevas mejoras:
 - Optimización general del sistema Android, la memoria y el rendimiento.
 - Integración del motor JavaScript V8 del Google Chrome en la aplicación Browser.
 - Lanzador de aplicaciones mejorado con accesos directos a las aplicaciones de teléfono y Browser.
 - Funcionalidad de Wi-Fi hotspot y tethering por USB.
 - Permite desactivar el tráfico de datos a través de la red del operador.
 - Actualización del Market con actualizaciones automática.

7. Gingerbread

- Versión 2.3
- Liberación del código el 6 de Diciembre del 2010
- Las nuevas mejoras se centran principalmente en la actualización del diseño de la interfaz. No obstante, otras fueron:
 - Soporte para pantallas extra grandes y resoluciones WXGA y mayores.
 - Soporte nativo para telefonía VoIP SIP
 - Soporte para reproducción de videos WebM/VP8 y decodificación de audio AAC
 - Nuevos efectos de audio como reverberación, ecualización, virtualización de los auriculares y refuerzo de graves

8. Honeycomb

- Versiones 3.0 / 3.1 / 3.2
- Está mejor soportado por las tablets.
- Las nuevas mejoras afectan a prácticamente todos los aspectos.
 - Escritorio 3D con widgets rediseñados.
 - Sistema multitarea mejorado.
 - Mejoras en navegador web y principales app del sistema operativo.
 - Soporte para videochat mediante google talk.
 - Mejor soporte de redes Wi-Fi así como guarda la configuración del SSID.
 - Añade soporte para una gran variedad de periféricos y accesorios con conexión USB: teclados, ratones, hubs, dispositivos de juego y cámaras digitales. Cuando un accesorio está conectado, el sistema busca la aplicación necesaria y ofrece su ejecución.
 - Los widgets pueden redimensionarse de forma manual sin la limitación del número de cuadros que tenga cada escritorio.
 - Se añade soporte opcional para redimensionar correctamente las aplicaciones inicialmente creadas para móvil para que se vean bien en Tablets

9. Ice Cream Sandwich

- Versión 4.0
- Las mejoras incluyen muchos cambios:
 - Unifica el uso de cualquier dispositivo, tanto teléfonos como tablets como televisores como netbooks.
 - La interfaz limpia y moderna, se llama "Holo".
 - Opción de utilizar los botones virtuales e la interfaz de usuario en lugar de los táctiles.
 - Llega la aceleración por hardware, lo que significa que la interfaz podrá ser manejada y dibujada por la GPU y aumentando notablemente su rapidez, su respuesta y evidentemente, la experiencia de usuario.
 - Multitarea mejorada, estilo Honeycomb. Añadiendo la posibilidad de finalizar una tarea simplemente desplazándola fuera de la lista.
 - Ha añadido un gestor del tráfico de datos de internet. El entorno le permite establecer alertas cuando llegue a una cierta cantidad de uso y desactivación de los datos cuando se pasa de su límite.
 - Los widgets están en una nueva pestaña del box de aplicaciones, que figuran en una lista similar a las aplicaciones en el menú principal. Por tanto la opción de ser añadidos mediante una larga pulsación en una zona vacía del escritorio ha desaparecido.

- El corrector de texto ha sido rediseñado y mejorado, ofreciendo la opción de tocar en una palabra para que nos aparezca una lista con las diferentes opciones de edición y sugerencias de palabras similares.

10. Jelly Bean

■ Versión 4.1

■ Principales mejoras:

- Mejora de la fluidez y de la estabilidad.
- Ajuste automático de widgets cuando se añaden al escritorio cambiando su tamaño y lugar.
- Dictado por voz mejorado con posibilidad de usarlo sin internet.
- Nuevas lenguas.
- Nuevos modos de acceso rápido al álbum de la cámara.
- Nueva función GoogleNow que nos permite tener una serie de tarjetas inteligentes que nos muestran información importante.

Estos datos han sido recogidos durante el período de 14 días que concluye el 2 de julio del 2012. Como se puede observar, la cuota de mercado de cada plataforma o versión del sistema operativo del robot es radicalmente distinta.

Ello, se debe a la relación de los dispositivos lanzados con dichas versiones y su éxito entre la población de una cierta zona.

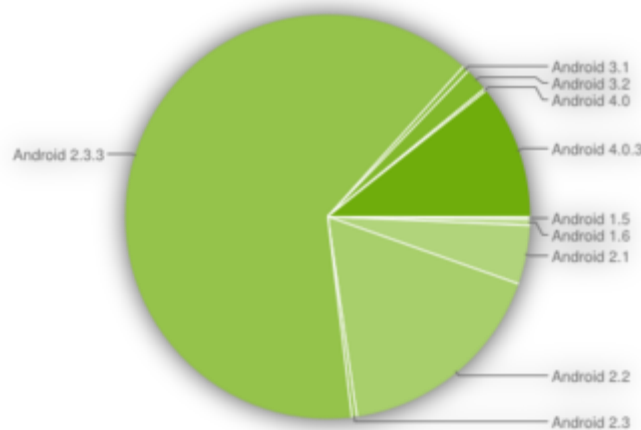


Ilustración 4: Cuota de mercado según la versión del sistema operativo.

Plataforma	Nivel de API	%
4.0.x <i>Ice Cream Sandwich</i>	14-15	10,9%
4.1.x <i>Jelly Bean</i>	16	¿¿??
3.x.x <i>Honeycomb</i>	11-13	2,4%
2.3.x <i>Gingerbread</i>	9-10	64,0%
2.2 <i>Froyo</i>	8	17,3%
2.1 <i>Eclair</i>	7	4,7%
1.6 <i>Donut</i>	4	0,5%
1.5 <i>CupcakeS</i>	3	0,2%

Tabla 1.: Cuota del mercado según las versiones del sistema operativo Android.

2.3.5.3 Arquitectura:

Los componentes principales del sistema operativo de Android son los siguientes):

- **Aplicaciones:** Las aplicaciones base incluyen un cliente de correo electrónico, programa de SMS, calendario, mapas, navegador, contactos y otros. Todas las aplicaciones están escritas en 'Java'.
- **Marco de trabajo de aplicaciones:** Los desarrolladores tienen acceso completo a la API del framework usados por las aplicaciones base. La arquitectura está diseñada para simplificar la reutilización y cualquier aplicación puede hacer uso de dichas capacidades.
- **Bibliotecas:** Android, incluye un conjunto de bibliotecas de C y C++ usadas por varios componentes del sistema. Estas características se exponen a los desarrolladores a través del marco de trabajo de aplicaciones de Android.
- **Runtime de Android:** Android, también incluye un set de bibliotecas base que proporcionan la mayor parte de las funciones disponibles en las bibliotecas base del lenguaje Java. Android corre su propio proceso, con su propia instancia de Dalvik, la máquina virtual usada. Dalvik, ha sido escrito de forma que un dispositivo podrá ser capaz de correr varias máquinas virtuales a la vez de forma eficiente. De este modo, Dalvik ejecutando archivos en el formato Dalvik Executable, el cual está optimizado para memoria mínima.

- **Núcleo Linux:** Para servicios base del sistema como la seguridad, la gestión de memoria o la gestión de procesos, Android, depende de Linux. El núcleo también actúa como una capa de abstracción entre el hardware y el resto de la pila de software.

2.3.5.4 Aplicaciones

Las aplicaciones se desarrollan habitualmente en el lenguaje 'Java' con el Software Development Kit de Android, SDK, de aquí en adelante. Pero están disponibles también otras herramientas de desarrollo para la programación de aplicaciones en C o en C++. Google App Inventor, y demás.

El desarrollo de aplicaciones para Android, no requiere aprender lenguajes complejos de programación, todo lo que se necesita, es un conocimiento aceptable de 'Java'. Además de tener un entorno de desarrollo adecuado (este autor recomienda el SDK de Android junto con el Eclipse).

Finalmente, apuntar que todas las aplicaciones están comprimidas en formato 'APK'. Éstas, se pueden instalar sin ninguna dificultad desde cualquier explorador de archivos.

El sistema operativo Android, principalmente es usado en la mayor parte de teléfonos inteligentes fabricados, no obstante, ésta, no es su única oportunidad de expandirse. Tablet, ordenadores portátiles netBooks, Google TV e incluso relojes de pulsera o auriculares, son otros de los dispositivos móviles para los que está pensado este sistema.

La plataforma hardware principal que soporta Android, es la arquitectura ARM. Existe soporte para x86 y Google TV utiliza una versión especial de Androidx86. El primer terminal móvil disponible para la venta al público fue el HTC Dream (Movistar), el cual vino de la mano con sus hermanos también en Android, de HTC Hero (Orange) y de HTC Magic (Vodafone) distribuidos en España por cada uno de los operadores principales.

2.4 Aplicaciones móviles de asistencia domótica.

Como se ha apuntado anteriormente, el crecimiento de las aplicaciones móviles en el mercado actual, es vertiginoso. Empresas y pequeños desarrolladores compiten por ideas originales e innovadoras que aporten a su producto ese toque especial que haga que sus aplicaciones triunfen.

Las aplicaciones móviles de la actualidad, tienen diversas características y un potencial enorme. Cosas como el streaming de audio o vídeo ya no son quimeras inalcanzables. Ya sea por el aumento de computación o por el ancho de banda disponible en dispositivos móviles, las aplicaciones cada vez son más ambiciosas y pretenden alcanzar servicios que tradicionalmente únicamente podía dar una computadora.

En el caso de nuestro sistema operativo, Android, las aplicaciones son distribuidas por medio de su 'Market' o 'Google Play', servicio creado para alojar las aplicaciones y posibilitar su descarga para todos los usuarios de dicha plataforma. En éste servicio es donde se especifica el coste de la aplicación para el usuario final. Puede ser gratuita o con cierto coser, normalmente mucho menor que en otras.

2.4.1 Ejemplos de aplicaciones domóticas:

- Securitas Direct:

Un ejemplo bastante sonado que ha dado mucho que hablar en los últimos tiempos, es la aplicación móvil de la empresa "Securitas Direct".

Dicha empresa, ofrece un servicio con un coste de 1€/mes con la contratación de sus alarmas, en las que te permiten tomar el control de tu alarma del hogar por medio de tu teléfono móvil inteligente. Esta aplicación te permitirá:

- Conectar/Desconectar tu alarma de forma remota.
- Conocer el estado de tu alarma (Saber si está conectada).
- Control de entradas y salidas. Puedes saber quién conecta y desconecta tu alarma, y a las horas que lo hace. (Personal de limpieza, empleados, etc).
- Puedes ver lo que ocurre en tu hogar o negocio cuando estás fuera. (Puedes pedir imágenes a tus Foto-detectores y visualizarlas desde la web de clientes privada).
- Abrir puertas de forma remota.

- Controlar tu domótica (Luces, calefacción, etc.).
- Ver la actividad de su alarma etc.



Ilustración 5: Ilustración de la aplicación de Securitas Direct.

- Crestron:

Otro buen ejemplo, es el de Crestron, una empresa fabricante de elementos domóticos para el hogar. Dicho fabricante de equipos de domótica, cuenta ya con aplicaciones para la gestión de sus sistemas en la plataforma Android.



Ilustración 6: Ilustración de la aplicación del fabricante Crestron.

El fabricante contaba ya con dos versiones en iOS y hace lo propio en el sistema operativo de Google. En 'Google Play' podrás encontrar dos versiones: la

primera, una versión gratuita (Crestron Mobile) y la segunda, una de pago (Crestron Mobile Pro), esta segunda por la friolera de 100 dólares (75,30 euros).

Las aplicaciones te permiten gestionar de forma remota (ya sea mediante una conexión WiFi o por 3G), los automatismos que tengas instalados en casa, o bien encender y apagar la luz o controlar la temperatura. Como vemos, son aplicaciones muy completas en cuanto a los dispositivos que controlan pero no tanto en el modo de controlarlos.

- ThinkFlood:

Finalmente, queremos destacar el trabajo del equipo de ThinkFlood en Android. Ésta compañía, conocida y galardonada, ha sorprendido con una gran apuesta tecnológica móvil actual. Google, ya avanzó que Android@Home junto con los 'Google Accessories', se encuentran en su proyecto para que todo dispositivo electrónico sea capaz de ser sincronizado con Android.

Por el momento, la aplicación que sincroniza su hardware, aun está en fase de pruebas. No obstante, ya se ha puesto un link para su descarga gratuita en su propia web para que cualquiera que le interese (desarrollador o particular) pueda bajársela y comprobar su increíble poder de sincronización "Todo en Uno".

Actualmente sigue siendo necesario disponer de un dispositivo iOS tal que iPhone, iPod Touch o iPad, para poder interactuar correctamente con sus controladores domóticos RedEyes, RedEyes Pro o RedEyes Mini (según sea la necesidad del usuario), pero la meta parece estar ya muy cerca.



Ilustración 7: Ilustración de la aplicación del fabricante ThinkFlood.

2.4.2 Competidores en la actualidad:

Hoy en día, no existe ningún competidor conocido que implemente el sistema domótico y regale una aplicación móvil con una interfaz adecuada para personas con discapacidad severa. Sin embargo, sí que existen proyectos que pretenden desarrollar ideas de domótica asistencial.

Se considera como un punto importante remarcar los posibles sistemas competidores que puedan compartir intereses con el proyecto que aquí se desarrolla.

Un primer ejemplo, puede ser el proyecto de 'ITADA'. Éste proyecto pretende a base de inteligencia domótica y una aplicación externa (no se especifica para qué tipo/s de dispositivo/s), conocer la necesidad del paciente y evitar que éste tenga que desplazarse hasta el hospital o centro sanitario. En la siguiente ilustración se pretende dar algo más de detalle del proyecto.

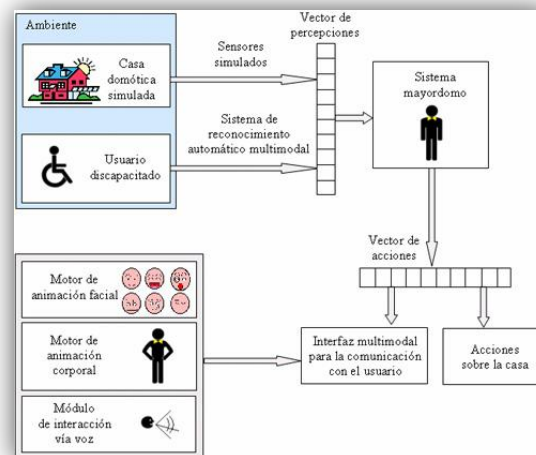


Ilustración 8: Diagrama del proyecto de domótica asistencial de 'ITADA'.

Otro proyecto de domótica asistencial, es el que llevan a cabo la UPM (Universidad Politécnica de Madrid) y la fundación de lucha contra el Alzheimer. En este proyecto, se centrarán en tres aspectos: ayudas técnicas para asistir al enfermo en su casa, ayudas técnicas que disminuyan la carga sobre la persona encargada de cuidar al paciente y finalmente ayudas tecnológicas que permitan mejorar la calidad de vida de ambos. Para cumplir con este objetivo se complementarán la experiencia de la FAE con los conocimientos técnicos del CeDInt.

Como se puede observar los proyectos de domótica asistencial más parecidos a la idea que aquí se desarrolla, están más enfocados a ayudar a las personas encargadas de cuidar o curar a los enfermos. Sin embargo, las ayudas no se centralizan directamente en el enfermo.

Finalmente, el proyecto ha desarrollar, se podría ver como un cúmulo de aspectos a implementar según los cuales, la aplicación funcionará con mayor o menor fluidez. Es por ello que la fase de pruebas y depuración, son, si cabe, más necesarias aún que en otros proyectos.

Además, existen una serie de servicios de Android necesarios para su correcto y completo funcionamiento.

3. Gestión del Proyecto.

En este apartado, se tratará de detallar la información relativa al proceso de gestión de tiempos y tareas.

3.1 Introducción.

Se pretende realizar una buena planificación del proyecto desde una primera instancia. Para ello, una buena estimación de tiempos en cada fase y sus tareas marcará el éxito de la planificación inicial.

Una mala estimación (no realista y sin valoraciones) de tiempos en tareas puede ser el punto de inflexión que determine si se cumplen los plazos o no a los que se compromete el equipo de desarrollo. Es decir, en muchos casos, la diferencia entre que el cliente termine satisfecho, o no.

Para determinar todas las estimaciones necesarias, se definirá en primer lugar, la metodología a usar en el proyecto, especificando sus fases y sus respectivas tareas. Seguidamente, se describirá el ciclo de vida del proyecto, y finalmente, se expondrá la organización del proyecto (recursos humanos) y la planificación total.

3.2 Metodología del proyecto.

Para este proyecto, la metodología adoptada es una métrica 3 con modificaciones para adecuar las fases a los requerimientos de nuestro proyecto pero con toda su esencia.

Esta métrica pretende alcanzar una serie de objetivos como proporcionar o definir sistemas de información que ayuden a conseguir los fines del cliente, mediante la definición de un marco estratégico de desarrollos de los mismos.

Dotar a los clientes, de productos software que satisfagan los requerimientos de los usuarios, dando un especial énfasis al análisis de requisitos. De éste modo, se tendrá la certeza de que el diseño y la implementación son lo suficientemente robustas.

Facilitar la comunicación entre los distintos participantes de la producción del software. El esquema a seguir según la metodología 3 adoptada por el equipo de desarrollo, es la siguiente:

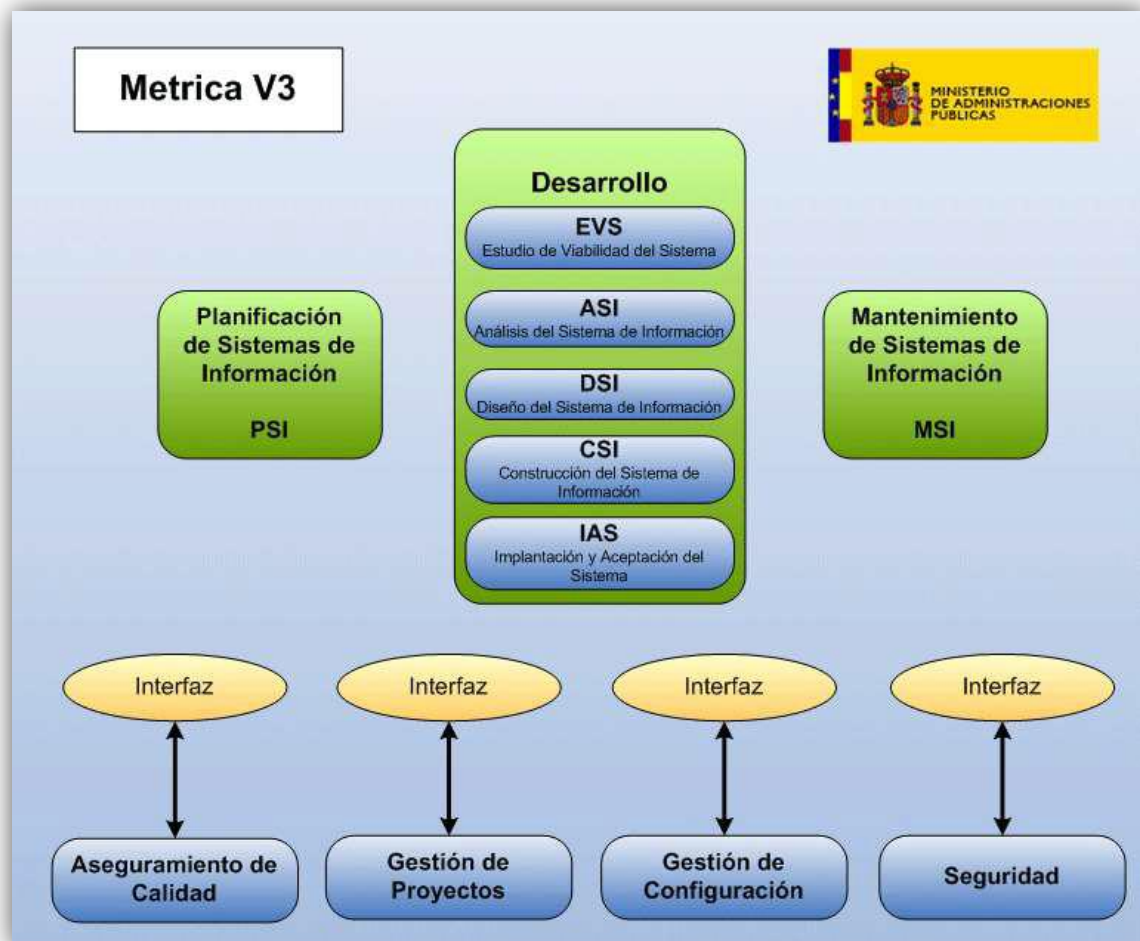


Ilustración 9: Métrica v3 ilustración proporcionada por el Ministerio de España ^[2].

3.2.1 Fases de la metodología adoptada:

- Análisis del Sistema de Información (ASI):

El objetivo de esta fase es la de recoger de forma detallada los requisitos de usuario y de software. De esta forma, se controla lo que el sistema de información debe cubrir.

Debe conseguir la especificación detallada del sistema de información, a través de un catálogo de requisitos y una serie de modelos que cubran las necesidades de información de los usuarios para los que se desarrollará el sistema de información y que serán la entrada para el proceso de Diseño del Sistema de Información (DSI).

Finalmente, decir, que debe obtener la definición de la arquitectura del sistema y del entorno tecnológico que le va a dar soporte, además de los componentes del sistema de información.

- Diseño del Sistema de Información (DSI):

El diseño de la arquitectura del sistema, dependerá de una gran variedad de las características del sistema. Es por ello que un buen trabajo en la fase ASI, simplificará mucho la tarea del DSI.

De este modo, tras esta tarea, se han de generar todas las especificaciones de construcción relativas al propio sistema, así como la especificación técnica del plan de pruebas, la definición de requisitos de implantación y el diseño de los procedimientos de migración y carga inicial de datos.

En nuestro caso, la migración de datos no será necesaria puesto que se trata de la primera versión del sistema, no obstante, se deberá tener en cuenta en futuras implementaciones de mejoras.

- Construcción del Sistema Informático (CSI):

A estas alturas del proyecto, para el inicio de esta fase, se toma como punto de partida los componentes diseñados previamente durante el proceso previo al de Diseño de Sistemas de Información (DSI).

Esta fase, debe cumplir con la construcción y prueba de los distintos componentes del sistema de información, de tal modo que a partir de un conjunto de especificaciones lógicas y físicas del mismo, se obtendrá el proceso de diseño del sistema de información.

- Implantación y Aceptación del Sistema (IAS):

Para el inicio de este proceso se toman como punto de partida los componentes del sistema probados. De esta forma, se deben haber probado de forma unitaria y se podrán integrar en el proceso de construcción del sistema de información (CSI).

Debe de realizar la entrega y aceptación del sistema en su totalidad y llevar a cabo las actividades oportunas para el paso a producción del sistema, en nuestro caso, al Market de Google. Además de la elaboración del plan de mantenimiento del sistema de forma que el responsable del mantenimiento lo conozca antes de que éste pase a producción.

Finalmente, decir que se ha desechado la fase de Mantenimiento del sistema definida en la métrica tres como fase final, Ello se ha debido a que éste equipo de desarrollo no iba a ser el encargado de llevarla a cabo, y se ha preferido delegar dicho trabajo a los que serán, sus futuros responsables. Además, comentar que toda la información de ésta métrica ha sido extraída de su documentación oficial.

3.3 Ciclo de vida.

El ciclo de vida de un producto software como el que se va a desarrollar, define el orden para las tareas o actividades a acometer. Ciertos desarrollos software requieren un tiempo y esfuerzo determinado que no siempre será igual, por ello, el orden en las tareas que se describirán a continuación, pueden variar.

Las etapas principales a acometer en el ciclo de vida software, son:

- **Análisis:** Construye un modelo de requisitos y expone los casos de uso y posibles escenarios de la aplicación.
- **Diseño:** A partir del modelo de análisis se deducen las estructuras de datos, la estructura en la que se ha de descomponer el sistema y las interfaces necesarias a implementar.
- **Codificación:** Construye el sistema. La salida de esta fase concluye en un código ejecutable.
- **Pruebas:** Se comprueba que se cumplen ciertos criterios de corrección y calidad.
- **Implantación:** Esta fase comprende la entrega del software en la plataforma destinada para el usuario.
- **Mantenimiento:** En esta fase, que tiene lugar después de la entrega del software al cliente se asegura que el sistema siga funcionando y adaptándose a nuevos requisitos.

En nuestro caso, las tareas de mantenimiento se terminarán por delegar en el equipo de expertos del laboratorio de domótica. No obstante, en un primer lugar, se tuvo que valorar el orden y la forma de adoptar cada una de las tareas descritas. Se valoraron distintas opciones como el ciclo de vida iterativo e incremental o los ciclos de vida en cascada, viéndose así, las debilidades de éste último sistema.

Mediante el modelo de desarrollo en cascada, se realizan las fases principales y más utilizadas de los ciclos de desarrollo del software, sin embargo, el cliente, recibe un único producto final. De este modo, surgen problemas como posibles errores en el desarrollo no detectados o simplemente ligeros cambios en configuraciones que no pueden ser aplicados *a posteriori*.

En nuestro caso, se optó por un ciclo de vida del software iterativo en el cual se irían incrementando las funcionalidades del sistema a medida que el cliente aprueba las distintas versiones del software. De este modo, en la primera iteración resultará una versión “Beta” de la aplicación, en la cual se mostrarán los servicios básicos de la aplicación.

Seguidamente y de forma incremental, se irán exponiendo distintas versiones de la aplicación del sistema domótico. Finalmente, se acabará exponiendo un software robusto y aprobado por el cliente.

La idea principal detrás del mejoramiento iterativo, reside en desarrollar un sistema de aplicaciones de manera incremental. De este modo, se permite al desarrollador sacar ventaja de lo que ha aprendido a lo largo del desarrollo de la versión anterior. El aprendizaje se divide en el desarrollo del sistema en sí y en su uso mediante las pruebas realizadas al final de cada fase de codificación.

En el modelo de ciclo de vida iterativo (el elegido por éste equipo de desarrollo) se pueden hacer cambios correctivos sobre el producto entregado. En cada iteración, se entrega al cliente una aplicación con ciertas partes del producto mejoradas. En la primera iteración, se podrá comprobar la aceptación del esquema fundamental y el flujo de la aplicación.

No obstante, cabe puntualizar que, a pesar de que en cada iteración se afrontan de nuevo todas las tareas del ciclo de vida del software, en algunos casos, se afrontará con mayor decisión de cambios, y en otras, sin tantos cambios. Por ejemplo, el análisis del sistema, en caso de que inicialmente haya sido bueno, no deberá ser modificado en exceso de cara a posteriores iteraciones. Así pasa lo mismo con fases como el diseño o las pruebas a implementar para cada codificación.

A continuación, se muestra una ilustración en la que se pretende explicar las fases y el orden de las que consta el ciclo de vida de este proyecto:

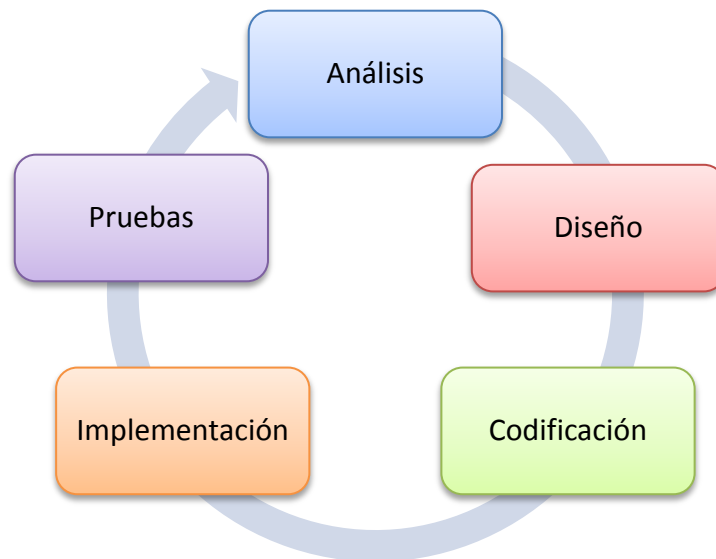


Ilustración 10: Ciclo de vida del proyecto.

3.4 Organización de Planificación.

La organización del proyecto, define los recursos humanos necesarios para llevar el desarrollo del proyecto al éxito. Para ello, se definen distintos roles dentro del proyecto para diferenciar las funciones, no obstante no se considerarán como figuras físicas independientes. Es decir, un sujeto podrá desempeñar más de una función dentro del proyecto.

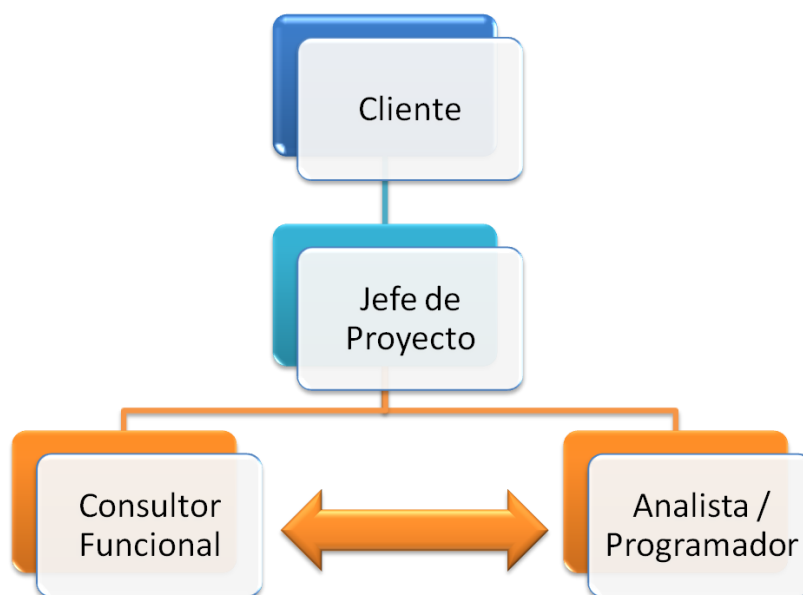


Ilustración 11: Organigrama del equipo de trabajo.

El nivel más alto jerárquicamente del proyecto es el cliente, el cual se identifica con la empresa en la que se integrará e implantará el módulo de reconocimiento auditivo destinado al sistema de domótica asistencial.

Seguidamente, se encuentra el jefe de proyecto, el cual, se identifica con el profesor/tutor encargado de dirigir las diferentes fases del proyecto y desempeñar el rol de responsable de que el equipo realice su trabajo. En este caso, Javier Fernández. De este modo, el jefe de proyecto es de perfil técnico y cumple con ciertas tareas extra como se especificará más adelante.

Finalmente, en el último nivel jerárquico, se encuentran dos perfiles esenciales en el proyecto.

En primer lugar, el perfil de consultar funcional, el cual, será el encargado de realizar las labores de consultoría del proyecto. En nuestro caso, se personifica en Jorge Alba. Cuenta con una adecuada formación en la extracción de requisitos. Aporta conocimientos en sistemas de aplicaciones móviles y sistemas domóticos.

Por último, se encuentra el perfil de analista y programador, el cual ha sido desempeñado por el autor de esta memoria de forma íntegra. Las competencias otorgadas han sido la de analizar e implementar el sistema de forma prácticamente íntegra, siendo ayudado en la primera por el tutor y jefe de proyecto, Javier Fernández.

3.6 Presupuesto.

Finalmente, en este apartado, se especifican los detalles de los costes que suponen el desarrollo del proyecto. Para ello, se han tomado referentes de costes extraídos de empresas similares a las que acometería un desarrollo de éstas características.

Se pretenden especificar todos los recursos necesarios empleados en la elaboración de éste proyecto. Para ello, es necesario realizar un cálculo de costes efectivo dado que será el presupuesto a aprobar por el cliente para implementar la solución diseñada.

Se procede a especificar los recursos necesarios, tanto materiales como humanos. En primer lugar, se expone el personal del que se dispondrá para el proyecto:

PERSONAL PROPIO O SUBCONTRATADO DEL QUE SE DISPONE	
NOMBRE Y DEDICACIÓN	COSTE
JAVIER FERNÁNDEZ – JEFE DE PROYECTO	2.300,00€
HORAS DEDICADAS AL PROYECTO 50€ *46 HORAS	
JORGE BLASCO ALÍS –CONSULTOR FUNCIONAL	665,00€
HORAS DEDICADAS AL PROYECTO 35€ *19 HORAS	
GONZALO HERRANZ – ANALISTA/PROGRAMADOR	10.000,00€
HORAS DEDICADAS AL PROYECTO 40 € *250 HORAS	
TOTAL	12.965,00€

Tabla 2.: Presupuesto de los recursos humanos.

Seguidamente, se exponen los recursos materiales de los que se requiere para la implementación del proyecto:

EQUIPOS NECESARIOS PARA EL ACOMETIMIENTO DEL PROYECTO					
<u>DESCRIPCIÓN</u>	<u>UNIDADES</u>	<u>COSTE UNIDAD</u>	<u>VIDA ÚTIL ESTIMADA</u>	<u>AMORTIZACIÓN</u>	<u>COSTE TOTAL</u>
Dispositivo Móvil Samsung Galaxy SII	1 ud	650,00€	24 meses/ud	27,10€/mes	94,85€
Portátiles Toshiba Satelite C66010H	2 uds	508,00€	18 meses/ud	56,60€/mes	198,10€
<u>TOTAL</u>					<u>292,95€</u>

Tabla 3.: Presupuesto de los materiales.

Finalmente, se exponen los costes del software necesario empleado:

SOFTWARE NECESARIOS PARA EL ACOMETIMIENTO DEL PROYECTO			
<u>DESCRIPCIÓN</u>	<u>UNIDADES</u>	<u>COSTE UNIDAD</u>	<u>COSTE TOTAL</u>
Windows 7 Home	1 UD	INC. EN TOSHIBA	0,00€
Microsoft Office Hogar y Pequeña empresa 2010	1 UD	379,00€	379,00€
SDK de Android 2.3.3	1 UD	GRATIS	0,00€
Eclipse Framework	1 UD	GRATIS	0,00€
Administrador de la BBDD SQLite	1 UD	GRATIS	0,00€
Google Translate	--	20\$/1MILL DE CARACTERES	--
<u>TOTAL</u>			<u>379,00€</u>

Tabla 4.: Presupuesto del software necesario.

Éstos, son todos los recursos necesarios para la elaboración del proyecto, por lo que el resultado total del presupuesto, será:

Recursos Humanos	12.960,00 €
Recursos Materiales	292,95 €
Recursos de Software	379,00 €

Tabla 5.: Presupuesto del software necesario.

Por lo que el presupuesto final del producto será de:

Total sin I.V.A.	13.631,95 €
I.V.A 21%	2.862,70 €
Total con I.V.A	16.494,65 €

Tabla 6.: Presupuesto del producto.

El cliente dispondrá de dos maneras para abonar el total de la cantidad acordada:

- Un único pago del total del presupuesto, con un descuento del 3% sobre la cantidad total. Dicha cantidad sería la siguiente: 91569,16 €
- Pago en 4 plazos, a acordar las fechas entre ambas partes, pagando en cada uno de ellos un 20%, 30%, 30% y un 20% de la cantidad total. La distribución de los pagos se muestra en la siguiente tabla.

Nº de Pago	Cantidad de Pago
1^{er} Pago	3.298,95 €
2^{do} Pago	4.948,40 €
3^{er} Pago	4.948,40 €
4º Pago	3.298,95 €

Tabla 7.: Presupuesto, formas de Pago.

Las cantidades anteriormente mencionadas llevan no incluidas un mantenimiento.

4. Análisis del Sistema.

En este apartado, se realizará un análisis en profundidad del sistema a desarrollar. Para ello, se comenzará por una introducción al sistema y seguidamente se describirán los requisitos del sistema. Éstos, se dividen principalmente en dos, los requisitos de usuario y de software.

Además, para finalizar el apartado, se expondrán los casos de uso del sistema. Definiendo todos los posibles escenarios de uso tanto de forma gráfica como textual, diferenciando dos tipos de usuario, en primer lugar, el usuario impedido y finalmente el usuario no impedido.

4.1 Introducción.

Este sistema, como ya se ha apuntado, se trata de un modulo de reconocimiento de voz para un sistema de domótica asistencial. Es decir, se trata del análisis, desarrollo e integración de dicho módulo auditivo en un sistema de domótica asistencial.

Por ello, el sistema a desarrollar, constará de un medio físico (micrófono) que puede estar incorporado o no, en el sistema móvil necesario. Además, dispondrá de una instalación de domótica asistencial en el hogar en el que se desee interactuar.

4.2 Requisitos del Sistema.

En primer lugar, se procederá a la extracción de requisitos del sistema. Éstos, se deberán diferenciar en dos tipos. En primer lugar, se definirán los requisitos de usuario, y seguidamente, los del software. Debido a que cada uno de ellos tiene un propósito se va a tratar de definir brevemente el de cada uno de ellos.

El requisito de usuario, pueden ser una descripción de lo que un sistema debe hacer. Este tipo de requisito, especifica lo que el sistema entregado debe ser capaz de realiza.

El requisito de software por otro lado, pretende especificar algo sobre el propio sistema y cómo ha de realizar las funciones deseadas.

La toma de requisitos o especificación de los requerimientos, se ha realizado mediante sesiones de análisis con el cliente y el equipo del laboratorio de domótica asistencial, que ha tenido experiencias con usuarios impedidos. De este modo se contrastan los diferentes perfiles que entran en juego y las necesidades de cada uno de cara a la implantación y utilización del sistema.

4.2.1 Requisitos de Usuario.

Como ya se ha apuntado en el sub-apartado anterior, estos requisitos son los encargados de delimitar, los qué debe hacer del sistema. De éste modo, dentro de éste tipo de requisitos se pueden distinguir dos tipos, el primero, el de “requisito de usuario de capacidad” (RUC), el segundo, el “requisito de usuario de restricción” (RUR).

Los requisitos de usuario de capacidad pretenden definir las funcionalidades del sistema. Es decir, delimita la capacidad de la aplicación.

Los requisitos de usuario de restricción pretenden definir las funcionalidades no necesarias en cuanto a del sistema. Es decir, restringe las funcionalidades para acotarlas a la funcionalidad deseada.

Para la descripción de cada uno de los requisitos, se utilizan los siguientes campos:

- **Identificador:** Identifica de forma unívoca cada uno de los requisitos de usuario que se explicarán a continuación. El formato para los requisitos de usuario de capacidad será RUC-XX, siendo XX la numeración del requisito. Del mismo modo, el formato para los requisitos de usuario de restricción, será RUR-XX.
- **Prioridad:** Indica la prioridad de implementación del requisito según el impacto en el desarrollo del mismo. Sus posibles valores, serán: “alta”, “media” o “baja”.
- **Fuente:** Definirá el origen del requisito, de donde surge la extracción del mismo. Sus valores posibles, será “cliente” o “expertos”.
- **Necesidad:** Indicará la necesidad de implementación del requisito. Sus posibles valores, serán “esencial”, “deseable” y “opcional”.
- **Estabilidad:** Dado que algunos requisitos pueden permanecer estables durante el ciclo completo de vida del software, se deben distinguir aquellos susceptibles de ser cambiados en un futuro próximo. En este caso, los posibles valores para éste atributo, será “estable” en caso de que no se prevea su cambio y “inestable” si estima que en un futuro podrá ser modificado cierto aspecto del sistema.
- **Descripción:** Expone brevemente el requisito.

Los requisitos vendrán representados por la siguiente plantilla:

Identificador		RUC/RUR - XX	
Prioridad	Alta/Media/Baja	Fuente	Cliente/Experto
Necesidad	Esencial Deseable/Opcional	Estabilidad	Estable/Inestable
Descripción	Descripción del requisito de usuario.		

Tabla 8.: Plantilla de Requisito de Usuario.

4.2.1.1 Requisitos de Capacidad.

Identificador		RUC - 01	
Prioridad	Alta	Fuente	Experto
Necesidad	Esencial	Estabilidad	Estable
Descripción	Contar con un dispositivo móvil.		

Tabla 9.: Requisito de Usuario de Capacidad 01.

Identificador		RUC – 02	
Prioridad	Media	Fuente	Experto
Necesidad	Deseable	Estabilidad	Estable
Descripción	Debe ser capaz de integrar un módulo para reconocer sistemas domóticos mediante etiquetas.		

Tabla 10.: Requisito de Usuario de Capacidad 02.

Identificador		RUC – 03	
Prioridad	Alta	Fuente	Experto
Necesidad	Esencial	Estabilidad	Estable
Descripción	Reconocer mandatos por reconocimiento de voz.		

Tabla 11.: Requisito de Usuario de Capacidad 03.

Identificador		RUC – 04	
Prioridad	Alta	Fuente	Experto
Necesidad	Esencial	Estabilidad	Estable
Descripción	Contar con un registro que contiene la información completa del sistema domótico.		

Tabla 12.: Requisito de Usuario de Capacidad 04.

Identificador		RUC – 05	
Prioridad	Alta	Fuente	Experto
Necesidad	Esencial		Estabilidad
Descripción	El usuario deberá ser capaz de usar todos los sistemas domóticos de la instalación.		

Tabla 13.: Requisito de Usuario de Capacidad 05.

Identificador		RUC – 06	
Prioridad	Alta	Fuente	Experto
Necesidad	Esencial		Estabilidad
Descripción	El sistema deberá ser portable.		

Tabla 14.: Requisito de Usuario de Capacidad 06.

Identificador		RUC – 07	
Prioridad	Alta	Fuente	Experto
Necesidad	Esencial		Estabilidad
Descripción	El sistema deberá reconocer únicamente objetos que se encuentren en el campo visual de la cámara.		

Tabla 15.: Requisito de Usuario de Capacidad 07.

Identificador		RUC – 08	
Prioridad	Alta	Fuente	Experto
Necesidad	Esencial		Estabilidad
Descripción	El sistema deberá reconocer únicamente acciones cuando previamente se halla reconocido un elemento.		

Tabla 16.: Requisito de Usuario de Capacidad 08.

Identificador		RUC – 09	
Prioridad	Alta	Fuente	Experto
Necesidad	Esencial		Estabilidad
Descripción	El dispositivo deberá ser capaz de emitir sonidos.		

Tabla 17.: Requisito de Usuario de Capacidad 09.

Identificador		RUC – 10	
Prioridad	Alta	Fuente	Experto
Necesidad	Esencial		Estabilidad
Descripción	El sistema deberá contar con conexión a la red.		

Tabla 18.: Requisito de Usuario de Capacidad 11.

Identificador		RUC – 11	
Prioridad	Media	Fuente	Cliente
Necesidad	Deseable	Estabilidad	Estable
Descripción	El sistema deberá ser adaptable a distintos idiomas.		

Tabla 19.: Requisito de Usuario de Capacidad 11.

Identificador		RUC – 12	
Prioridad	Alta	Fuente	Experto
Necesidad	Esencial	Estabilidad	Estable
Descripción	El sistema deberá contar con distintos modos, tanto automático como manual.		

Tabla 20.: Requisito de Usuario de Capacidad 12.

Identificador		RUC – 13	
Prioridad	Alta	Fuente	Experto
Necesidad	Esencial	Estabilidad	Estable
Descripción	El sistema deberá poder enviar peticiones al servidor domótico.		

Tabla 21.: Requisito de Usuario de Capacidad 13.

Identificador		RUC – 14	
Prioridad	Alta	Fuente	Experto
Necesidad	Esencial	Estabilidad	Estable
Descripción	El servidor domótico deberá poder recibir peticiones tipo 'HTTP' y contestarlas.		

Tabla 22.: Requisito de Usuario de Capacidad 14.

Identificador		RUC – 15	
Prioridad	Alta	Fuente	Experto
Necesidad	Esencial	Estabilidad	Estable
Descripción	El servidor domótico deberá poder lanzar mandatos a los distintos dispositivos de la red.		

Tabla 23.: Requisito de Usuario de Capacidad 15.

Identificador		RUC – 16	
Prioridad	Alta	Fuente	Experto
Necesidad	Esencial	Estabilidad	Estable
Descripción	Los dispositivos IP deberán estar integrados en la misma red que el servidor domótico.		

Tabla 24.: Requisito de Usuario de Capacidad 16.

Identificador		RUC – 17	
Prioridad	Alta	Fuente	Experto
Necesidad	Esencial		Estabilidad
Descripción	Los dispositivos de la red deberán contener acciones mecanizadas.		

Tabla 25.: Requisito de Usuario de Capacidad 17.

4.2.1.2 Requisitos de Restricción.

Identificador		RUR – 01	
Prioridad	Alta	Fuente	Experto
Necesidad	Esencial		Estabilidad
Descripción	No debe ser necesaria la interacción con la pantalla en al menos un modo del sistema.		

Tabla 26.: Requisito de Usuario de Restricción 01.

Identificador		RUR – 02	
Prioridad	Alta	Fuente	Cliente
Necesidad	Esencial		Estabilidad
Descripción	No debe ser necesario parametrizar el lenguaje del sistema.		

Tabla 27.: Requisito de Usuario de Restricción 02.

Identificador		RUR – 03	
Prioridad	Alta	Fuente	Cliente
Necesidad	Esencial		Estabilidad
Descripción	No debe ser necesario el conocimiento de cada posible acción para cada sistema domótico.		

Tabla 28.: Requisito de Usuario de Restricción 03.

4.2.2 Requisitos del Software.

Seguidamente, llegamos a los requisitos del Software, que como se ha definido anteriormente, son los encargados de definir el XXX del sistema. Éste tipo de requisitos, de nuevo, se dividen en dos tipos, los “requisitos de software funcionales” (RSF) y los “requisitos de software no funcionales” (RSNF).

Los requisitos de software de funcionales, pretenden definir el comportamiento interno del software, detalles técnicos, manipulación de datos y otras funcionalidades del sistema.

Los requisitos de software no funcionales, se enfocan al diseño o la implementación del sistema.

El formato de la tabla de requisitos de software tanto funcional como no funcional, se mantiene idéntica a la anterior. Sin embargo, los identificadores cambian de formato, de tal modo que quedan como:

- **RSF – XX:** Representa el identificador para los Requisitos de Software Funcionales. Además el campo XX identifica el valor numérico del requisito, el cual será único.
- **RSNF – XX:** Representa el identificador para los Requisitos de Software No Funcionales. Además el campo XX identifica el valor numérico del requisito, el cual será único.

Finalmente, decir que para éste tipo de requisitos la fuente será un Requisito de Usuario de, tanto de Capacidad como de Restricción (RUR o RUC). Además, cabe añadir que varios Requisitos de Software (RSF o RSNF) pueden corresponder con un único Requisito de Usuario (RUR o RUC)

4.2.2.1 Requisitos Funcionales.

Identificador		RSF – 01	
Prioridad	Alta	Fuente	RUC-01
Necesidad	Esencial	Estabilidad	Estable
Descripción	Se contará con un dispositivo móvil con sistema operativo Android integrado.		

Tabla 29.: Requisito de Software funcional 01.

Identificador RSF – 02			
Prioridad	Alta	Fuente	RUC-01
Necesidad	Esencial	Estabilidad	Estable
Descripción	Se contará con un dispositivo móvil Android de la versión mínima 2.3.3.		

Tabla 30.: Requisito de Software funcional 02.

Identificador RSF – 03			
Prioridad	Alta	Fuente	RUC-01
Necesidad	Esencial	Estabilidad	Estable
Descripción	Se contará con un dispositivo móvil con cámara y micrófono, éstos pueden estar integrados o periféricos		

Tabla 31.: Requisito de Software funcional 03.

Identificador RSF – 04			
Prioridad	Alta	Fuente	RUC-02
Necesidad	Esencial	Estabilidad	Estable
Descripción	Se podrá integrar un módulo que reconocerá los elementos domóticos mediante etiquetas por medio de un módulo óptico de códigos tipo QR.		

Tabla 32.: Requisito de Software funcional 04.

Identificador RSF – 05			
Prioridad	Media	Fuente	RUC-02
Necesidad	Deseable	Estabilidad	Estable
Descripción	Se reconocerán los elementos domóticos mediante mandatos por voz por medio de un módulo de reconocimiento de voz.		

Tabla 33.: Requisito de Software funcional 05.

Identificador RSF – 06			
Prioridad	Media	Fuente	RUC-02
Necesidad	Deseable	Estabilidad	Estable
Descripción	Se reconocerán la selección de elementos domóticos mediante la selección manual, de forma adicional.		

Tabla 34.: Requisito de Software funcional 06.

Identificador		RSF – 07	
Prioridad	Alta	Fuente	RUC-03
Necesidad	Esencial	Estabilidad	Estable
Descripción	Se deberán poder reconocer las acciones para el elemento domótico previamente seleccionado, por medio del módulo de reconocimiento de voz.		

Tabla 35.: Requisito de Software funcional 07.

Identificador		RSF – 08	
Prioridad	Media	Fuente	RUC-03
Necesidad	Deseable	Estabilidad	Estable
Descripción	Se deberán poder reconocer los mandatos para el elemento domótico previamente seleccionado, por medio de la selección táctil de forma adicional.		

Tabla 36.: Requisito de Software funcional 08.

Identificador		RSF – 09	
Prioridad	Alta	Fuente	RUC-04
Necesidad	Esencial	Estabilidad	Estable
Descripción	Se deberán almacenar en el dispositivo los elementos del sistema domótico.		

Tabla 37.: Requisito de Software funcional 09.

Identificador		RSF – 10	
Prioridad	Alta	Fuente	RUC-04
Necesidad	Esencial	Estabilidad	Estable
Descripción	Se deberán almacenar en el dispositivo los mandatos posibles de cada elemento del sistema domótico.		

Tabla 38.: Requisito de Software funcional 10.

Identificador		RSF – 11	
Prioridad	Alta	Fuente	RUC-04
Necesidad	Esencial	Estabilidad	Estable
Descripción	Se deberán almacenar en el dispositivo las URL para los mandatos posibles de cada elemento del sistema domótico.		

Tabla 39.: Requisito de Software funcional 11.

Identificador		RSF – 12	
Prioridad	Alta	Fuente	RUC-04
Necesidad	Esencial	Estabilidad	Estable
Descripción	Se deberán almacenar en el dispositivo los mandatos posibles de cada elemento relacionados con el elemento del sistema domótico que realiza dicha acción. Así mismo tiene que relacionarlos con la URL a mandar al servidor.		

Tabla 40.: Requisito de Software funcional 12.

Identificador		RSF – 13	
Prioridad	Alta	Fuente	RUC-04
Necesidad	Esencial	Estabilidad	Estable
Descripción	El sistema deberá de ser capaz de ejecutar consultas a la base de datos del dispositivo.		

Tabla 41.: Requisito de Software funcional 13.

Identificador		RSF – 14	
Prioridad	Alta	Fuente	RUC-04
Necesidad	Esencial	Estabilidad	Estable
Descripción	El sistema deberá de ser capaz de relacionar los mandatos recibidos por voz con las acciones resultantes de la consulta sobre la base de datos.		

Tabla 42.: Requisito de Software funcional 14.

Identificador		RSF – 15	
Prioridad	Alta	Fuente	RUC-04
Necesidad	Esencial	Estabilidad	Estable
Descripción	El sistema deberá de ser capaz de relacionar el resultado de comparar los mandatos obtenidos por voz con los resultados de la consulta de acciones según el elemento previamente identificado.		

Tabla 43.: Requisito de Software funcional 15.

Identificador		RSF – 16	
Prioridad	Alta	Fuente	RUC-05
Necesidad	Esencial	Estabilidad	Estable
Descripción	El sistema deberá de ser capaz de diferenciar los distintos elementos del sistema domótico.		

Tabla 44.: Requisito de Software funcional 16.

Identificador RSF – 17			
Prioridad	Alta	Fuente	RUC-05
Necesidad	Esencial	Estabilidad	Estable
Descripción	El sistema deberá de ser capaz de relacionar los distintos elementos del sistema domótico y sus posibles acciones.		

Tabla 45.: Requisito de Software funcional 17.

Identificador RSF – 18			
Prioridad	Alta	Fuente	RUC-05
Necesidad	Esencial	Estabilidad	Estable
Descripción	El sistema deberá de tener acceso a todos los elementos domóticos del sistema.		

Tabla 46.: Requisito de Software funcional 18.

Identificador RSF – 19			
Prioridad	Alta	Fuente	RUC-06
Necesidad	Esencial	Estabilidad	Estable
Descripción	El sistema deberá poder ser instalado en el dispositivo móvil.		

Tabla 47.: Requisito de Software funcional 19.

Identificador RSF – 20			
Prioridad	Alta	Fuente	RUC-06
Necesidad	Esencial	Estabilidad	Estable
Descripción	El sistema deberá poder ser energéticamente autónomo.		

Tabla 48.: Requisito de Software funcional 20.

Identificador RSF – 21			
Prioridad	Alta	Fuente	RUC-07
Necesidad	Esencial	Estabilidad	Estable
Descripción	El sistema deberá poder integrar un módulo de reconocimiento visual capaz de reconocer mediante una cámara las etiquetas de tipo QR del elemento domótico.		

Tabla 49.: Requisito de Software funcional 21.

Identificador RSF – 22			
Prioridad	Alta	Fuente	RUC-07
Necesidad	Esencial	Estabilidad	Estable
Descripción	El sistema deberá poder integrar un módulo que deseche etiquetas QR cuando no se fije la cámara un tiempo de 2 segundos sobre él.		

Tabla 50.: Requisito de Software funcional 22.

Identificador RSF – 23			
Prioridad	Alta	Fuente	RUC-08
Necesidad	Esencial	Estabilidad	Estable
Descripción	El sistema deberá poder reconocer mandatos por voz en el momento en que se haya reconocido un elemento anteriormente.		

Tabla 51.: Requisito de Software funcional 23.

Identificador RSF – 24			
Prioridad	Alta	Fuente	RUC-08
Necesidad	Esencial	Estabilidad	Estable
Descripción	El sistema deberá poder desechar acciones no comprendidas para el elemento.		

Tabla 52.: Requisito de Software funcional 24.

Identificador RSF – 25			
Prioridad	Alta	Fuente	RUC-08
Necesidad	Esencial	Estabilidad	Estable
Descripción	El sistema deberá poder re-intentar el reconocimiento por voz de la acción en caso de no haber sido reconocida anteriormente.		

Tabla 53.: Requisito de Software funcional 25.

Identificador RSF – 26			
Prioridad	Media	Fuente	RUC-09
Necesidad	Deseable	Estabilidad	Estable
Descripción	El sistema deberá poder reproducir texto de forma sonora.		

Tabla 54.: Requisito de Software funcional 26.

Identificador		RSF – 27	
Prioridad	Media	Fuente	RUC-09
Necesidad	Deseable	Estabilidad	Estable
Descripción	El sistema deberá poder reproducir alarmas sonoras en caso de error.		

Tabla 55.: Requisito de Software funcional 27.

Identificador		RSF – 28	
Prioridad	Media	Fuente	RUC-10
Necesidad	Deseable	Estabilidad	Estable
Descripción	El sistema deberá tener conexión a la red donde se encuentre el servidor domótico.		

Tabla 56.: Requisito de Software funcional 28.

Identificador		RSF – 29	
Prioridad	Media	Fuente	RUC-10
Necesidad	Deseable	Estabilidad	Estable
Descripción	El sistema deberá tener conexión a internet.		

Tabla 57.: Requisito de Software funcional 29.

Identificador		RSF – 30	
Prioridad	Media	Fuente	RUC-11
Necesidad	Deseable	Estabilidad	Estable
Descripción	El sistema deberá poder reconocer distintos idiomas del sistema operativo Android de forma automática.		

Tabla 58.: Requisito de Software funcional 30.

Identificador		RSF – 31	
Prioridad	Media	Fuente	RUC-11
Necesidad	Deseable	Estabilidad	Estable
Descripción	El sistema deberá poder reproducir el texto de los avisos sonoros en distintos idiomas de forma automática.		

Tabla 59.: Requisito de Software funcional 31.

Identificador		RSF – 32	
Prioridad	Media	Fuente	RUC-12
Necesidad	Deseable	Estabilidad	Estable
Descripción	El sistema deberá tener un modo manual que permita la selección táctil tanto del elemento como de la acción.		

Tabla 60.: Requisito de Software funcional 32.

Identificador		RSF – 33	
Prioridad	Alta	Fuente	RUC-12
Necesidad	Esencial	Estabilidad	Estable
Descripción	El sistema deberá tener un modo automático que permita la selección de elemento y acciones de forma automática.		

Tabla 61.: Requisito de Software funcional 33.

Identificador		RSF – 34	
Prioridad	Alta	Fuente	RUC-13
Necesidad	Esencial	Estabilidad	Estable
Descripción	El sistema deberá tener la instalación del protocolo HTTP para poder realizar peticiones sobre el servidor de su red.		

Tabla 62.: Requisito de Software funcional 34.

Identificador		RSF – 35	
Prioridad	Alta	Fuente	RUC-14
Necesidad	Esencial	Estabilidad	Estable
Descripción	El servidor domótico deberá tener un Servlet escuchando en un cierto puerto para tratar las posibles peticiones HTTP request.		

Tabla 63.: Requisito de Software funcional 35.

Identificador		RSF – 36	
Prioridad	Alta	Fuente	RUC-14
Necesidad	Esencial	Estabilidad	Estable
Descripción	El servidor domótico deberá contestar la petición realizada sobre el mismo al dispositivo origen con un mensaje de confirmación o de error.		

Tabla 64.: Requisito de Software funcional 36.

Identificador		RSF – 37	
Prioridad	Alta	Fuente	RUC-15
Necesidad	Esencial	Estabilidad	Estable
Descripción	El servidor domótico deberá poder lanzar peticiones sobre los elementos IP, deberá tener el control sobre todos los elementos.		

Tabla 65.: Requisito de Software funcional 37.

Identificador		RSF – 38	
Prioridad	Alta	Fuente	RUC-16
Necesidad	Esencial	Estabilidad	Estable
Descripción	El sistema deberá contener una red interna que integre los dispositivos IP, el servidor domótico y la aplicación móvil.		

Tabla 66.: Requisito de Software funcional 38.

Identificador		RSF – 39	
Prioridad	Alta	Fuente	RUC-17
Necesidad	Esencial	Estabilidad	Estable
Descripción	Los dispositivos IP deberán ser capaces de automatizar las acciones de los mandatos recibidos.		

Tabla 67.: Requisito de Software funcional 39.

Identificador		RSF – 40	
Prioridad	Alta	Fuente	RUC-17
Necesidad	Esencial	Estabilidad	Estable
Descripción	Los dispositivos IP deberán ser capaces de contestar el mandato del servidor con un mensaje de éxito o de error.		

Tabla 68.: Requisito de Software funcional 40.

4.2.2.2 Requisitos No Funcionales.

Identificador		RSNF – 01	
Prioridad	Alta	Fuente	RUR-01
Necesidad	Esencial	Estabilidad	Estable
Descripción	El usuario no debe interactuar con el medio táctil del dispositivo para seleccionar un elemento.		

Tabla 69.: Requisito de Software no funcional 01.

Identificador		RSNF – 02	
Prioridad	Alta	Fuente	RUR-01
Necesidad	Esencial	Estabilidad	Estable
Descripción	El usuario no debe interactuar con el medio táctil del dispositivo para seleccionar una acción.		

Tabla 70.: Requisito de Software no funcional 02.

Identificador		RSNF – 03	
Prioridad	Alta	Fuente	RUR-01
Necesidad	Esencial	Estabilidad	Estable
Descripción	El usuario no debe interactuar con el medio táctil del dispositivo para seleccionar repetir el reconocimiento de un mandato no identificado.		

Tabla 71.: Requisito de Software no funcional 03.

Identificador		RSNF – 04	
Prioridad	Alta	Fuente	RUR-01
Necesidad	Esencial	Estabilidad	Estable
Descripción	El usuario no debe interactuar con el medio táctil del dispositivo para seleccionar repetir el reconocimiento de un mandato no identificado.		

Tabla 72.: Requisito de Software no funcional 04.

Identificador		RSNF – 05	
Prioridad	Alta	Fuente	RUR-02
Necesidad	Esencial	Estabilidad	Estable
Descripción	El usuario no debe interactuar con el dispositivo para seleccionar el lenguaje de la aplicación.		

Tabla 73.: Requisito de Software no funcional 05.

Identificador		RSNF – 06	
Prioridad	Alta	Fuente	RUR-03
Necesidad	Esencial	Estabilidad	Estable
Descripción	El usuario no debe tener conocimiento de las posibles acciones de cada elemento. El sistema ha de comunicar al usuario las acciones por medio sonoro.		

Tabla 74.: Requisito de Software no funcional 06.

4.2.3 Matriz de trazabilidad de requisitos de usuario y requisitos del software.

La matriz de trazabilidad de requisitos de usuario y de software, pretenden mostrar una correlación entre ellos. En caso de que éstos, se correlacionen, quedará patente la robustez del sistema. En primer lugar, se expone la matriz de requisitos de software funcionales (RSF) frente a los requisitos de usuario de capacidad (RUC).

Requisitos Software \ Requisitos Usuario	RSF - 01	RSF - 02	RSF - 03	RSF - 04	RSF - 05	RSF - 06	RSF - 07	RSF - 08	RSF - 09	RSF - 10	RSF - 11	RSF - 12	RSF - 13	RSF - 14	RSF - 15	RSF - 16	RSF - 17	RSF - 18	RSF - 19	RSF - 20	RSF - 21	RSF - 22	RSF - 23	RSF - 24	RSF - 25	RSF - 26	RSF - 27	RSF - 28	RSF - 29	RSF - 30	RSF - 31	RSF - 32	RSF - 33	RSF - 34	RSF - 35	RSF - 36	RSF - 37	RSF - 38	RSF - 39	RSF - 40
RUC - 01	X	X	X																																					
RUC - 02				X	X	X																																		
RUC - 03							X	X																																
RUC - 04									X	X	X	X	X	X	X																									
RUC - 05																X	X	X																						
RUC - 06																			X	X																				
RUC - 07																					X	X																		
RUC - 08																							X	X	X															
RUC - 09																										X	X													
RUC - 10																												X	X											
RUC - 11																														X	X									
RUC - 12																																X	X							
RUC - 13																																		X						
RUC - 14																																			X	X				
RUC - 15																																					X			
RUC - 16																																							X	
RUC - 17																																							X	X

Tabla 75.: Matriz de trazabilidad de requisitos de usuario de capacidad y requisitos de software funcionales.

En segundo lugar, se expone la matriz de requisitos de software no funcionales (RSNF) frente a los requisitos de usuario de restricción (RUR).

Requisitos Software Requisitos Usuario	RSNF – 01	RSNF – 02	RSNF – 03	RSNF – 04	RSNF – 05	RSNF – 06
RUR – 01	X	X	X	X		
RUR – 02					X	
RUR – 03						X

Tabla 76.: Matriz de trazabilidad de requisitos de usuario de restricción y requisitos no funcionales.

4.3 Casos de Uso.

En este apartado, se detallan los casos de uso que van a ser contemplados para el sistema de reconocimiento de voz para el módulo de domótica asistencial. Por lo que se pretenden exponer y describir tanto gráfica como textualmente todos los posibles escenarios que se pueden dar en la aplicación.

Para cubrir todos los posibles casos, como se ha apuntado anteriormente, se distinguirán dos tipos de usuarios, el usuario impedido, y el no impedido. De este modo, se pretende diferenciar dos tipos de uso para los que se pretende dar soporte.

Por tanto, mediante el método de casos de uso describe las iteraciones típicas usuario-sistema y cada caso que se define es una alternativa de uso y por tanto una forma de usar el sistema. Se podrá contar con pequeñas modificaciones en el momento de la implementación, no obstante se pretende que éstos, sean los definitivos.

4.3.1 Diagramas de casos de uso.

En este subapartado, se presentan los diagramas de casos de uso del sistema. Además, dichos casos de uso, representan todos los casos del sistema incluyendo distinciones entre los dos principales usuarios que interactuarán con el sistema. Por lo que se pretenden cubrir todos los posibles escenarios de la aplicación en éste apartado.

El primer grupo de los diagramas, representa el uso que hará un usuario impedido o de movilidad reducida. En el segundo de los diagramas, se representa el uso que, podrá realizar un usuario sin movilidad reducida.

4.3.1.1 Diagramas de caso de uso – Usuario impedido.

Los casos de uso de un usuario discapacitado o impedido serán:

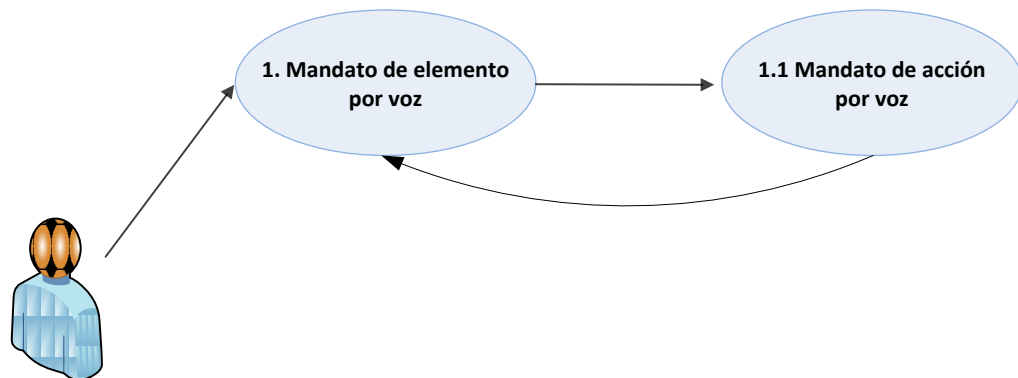


Ilustración 15: Diagrama de casos de uso de un usuario impedido.

4.3.1.2 Diagramas de caso de uso – Usuario no impedido.

Los casos de uso de un usuario no impedido, serán:

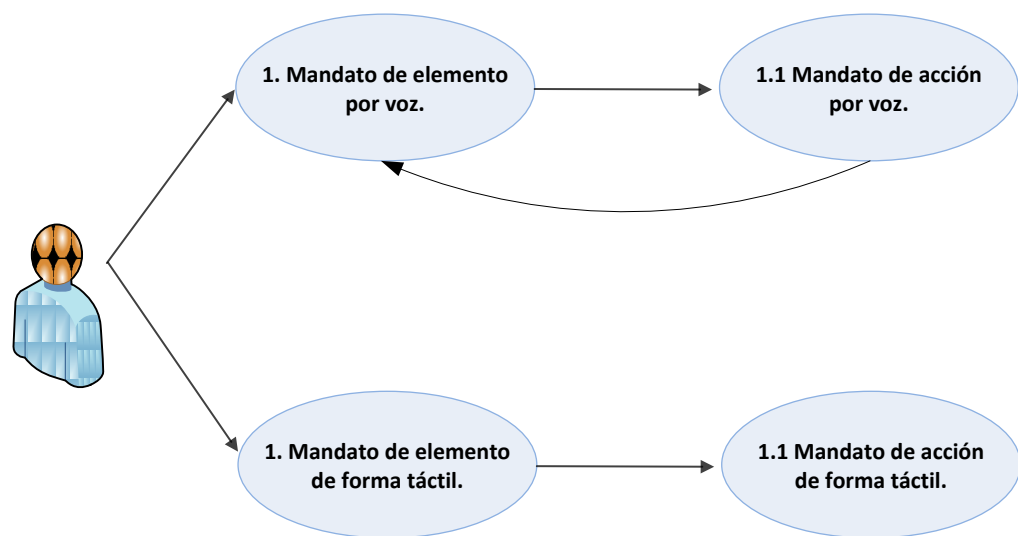


Ilustración 16: Diagrama de casos de uso de un usuario no impedido.

4.3.2 Descripción textual de los casos de uso.

En éste sub-apartado, se pretende realizar una descripción de los casos de uso dispuestos gráficamente con anterioridad. De éste modo, se pretende que el lector se familiarice más con el uso de la aplicación y los posibles escenarios que se pueden tener que afrontar.

Para la descripción de los casos de uso de cada uno de los tipos de actores, se utilizará la siguiente plantilla.

CU – XX	Descripción del caso de uso
Actores	
Objeto	
Precondiciones	
Postcondiciones	
Escenario	
Escenario Alternativo	

Tabla 77.: Plantilla de descripción de los casos de uso.

Los campos de esta plantilla definen los casos de uso, por lo que conviene explicarlos brevemente:

- **Identificador:** Identifica de forma unívoca cada uno de casos de uso, de este modo, se facilitarán las posibles referencias al mismo. El formato de este campo será **CU – X.X**. Consta de un primer prefijo CU, seguido de **X.X** dos dígitos separados por un punto. Estos dígitos indicarán la versión del requisito y su variante.
- **Descripción:** Se trata de una breve descripción del caso de uso.
- **Actores:** Son los agentes externos que interactúan con el sistema, de este modo, cada actor representará un posible rol dentro de la aplicación. En nuestro caso serán dos posibles valores “Usuario Impedido”, “Usuario no Impedido”.
- **Objetivo:** Finalidad del caso de uso o meta que se pretende alcanzar mediante la funcionalidad que describe.
- **Precondiciones:** Estado en el que se encuentra el sistema en el momento previo a la ejecución del caso de uso que se describe.
- **Postcondiciones:** Estado que se alcanza después de la ejecución del caso de uso. Es decir cómo termina el sistema la ejecución de éste caso de uso.
- **Escenario Principal:** Secuencia de acciones principales de la interacción en el escenario básico. Se detallará la información intercambiada, acciones del sistema y cambios observados en el sistema.
- **Escenarios Alternativos:** Secuencias de acciones alternativas respecto a cierto/s punto/s del escenario principal.

4.3.2.1 Descripción de caso de uso – Usuario impedido.

Éste tipo de usuario, es el principal protagonista de nuestros usuarios. Se pretende que el sistema dé soporte principalmente a este tipo de usuarios. Para ello, la usabilidad de la aplicación, ha de ser total.

CU – 1.0 Mandato de elemento por voz.	
Actores	Usuario Impedido / Usuario No Impedido
Objeto	Elección de un elemento del sistema domótico por voz.
Precondiciones	El usuario no ha elegido un elemento del sistema domótico en la acción anterior.
Postcondiciones	El elemento queda seleccionado y el sistema de reconocimiento de voz pendiente de que el usuario elija una opción.
Escenario	<ol style="list-style-type: none"> 1. Seleccionar modo automático de la aplicación. 2. La aplicación inicia de forma automático el reconocimiento de voz. 3. El usuario dice un elemento del sistema domótico. 4. El sistema reconoce el elemento domótico e inicia la siguiente tarea.
Escenario Alternativo	<p>4a. El sistema no logra reconocer el mandato indicado por voz.</p> <p>4b. El sistema inicia de nuevo el reconocimiento por voz.</p>

Tabla 78.: CU-1.0 Mandato de elemento por voz.

CU –1.1 Mandato de acción por voz.	
Actores	Usuario Impedido / Usuario No Impedido
Objeto	Elección de una acción del sistema domótico por voz.
Precondiciones	El usuario ha elegido un elemento del sistema domótico en la acción anterior.
Postcondiciones	La acción es reconocida y se envía la petición al servidor.
Escenario	<ol style="list-style-type: none"> 1. Seleccionar modo automático de la aplicación. 2. La aplicación inicia de forma automático el reconocimiento de voz. 3. El usuario dice un elemento del sistema domótico. 4. El sistema reconoce el elemento domótico e inicia la siguiente tarea. 5. El sistema inicia la búsqueda de las acciones posibles del elemento seleccionado y las

	<p>muestra por pantalla.</p> <p>6. El sistema iniciar el reconocimiento por voz de la acción a realizar.</p> <p>7. El sistema realiza el mandato al servidor de la acción reconocida de forma automática.</p> <p>8. El sistema devuelve el control al caso de uso anterior y espera un nuevo elemento del sistema.</p>
Escenario Alternativo	<p>4a. El sistema no logra reconocer el elemento indicado por voz.</p> <p>4b. El sistema inicia de nuevo el reconocimiento por voz.</p> <p>6a. El sistema no logra reconocer el mandato indicado por voz.</p> <p>6b. El sistema inicia de nuevo el reconocimiento por voz.</p>

Tabla 79.: CU-1.1 Mandato de elemento por voz.

4.3.2.2 Descripción de caso de uso – Usuario no impedido (Extra).

Éste usuario, se considera una funcionalidad extra, no obstante, es el que más casos de uso tiene dado que los dos anteriores, aplicables a una persona impedida pueden ser también desempeñadas por éste tipo de actores. Dado que los dos casos de uso anteriores pueden ser aplicables a éste tipo de actores, no se repetirán de nuevo y únicamente se expondrán sus dos casos de uso exclusivos:

CU – 2.0 Mandato de elemento por voz.	
Actores	Usuario No Impedido
Objeto	Elección de un elemento del sistema domótico por selección táctil.
Precondiciones	El usuario no ha elegido un elemento del sistema domótico en la acción anterior.
Postcondiciones	El elemento queda seleccionado y el sistema muestra la siguiente interfaz con las acciones posibles para que el actor elija una opción.
Escenario	<ol style="list-style-type: none"> 1. Seleccionar modo manual de la aplicación. 2. La aplicación muestra por pantalla una lista de los elementos del sistema domótico. 3. El usuario selecciona un elemento del sistema domótico. 4. El sistema selecciona el elemento domótico e inicia la siguiente tarea.
Escenario Alternativo	--

Tabla 80.: CU-2.0 Mandato de elemento por selección táctil.

CU –2.1 Mandato de acción por voz.	
Actores	Usuario No Impedido
Objeto	Elección de una acción del sistema domótico por voz.
Precondiciones	El usuario ha elegido un elemento del sistema domótico en la acción anterior.
Postcondiciones	La acción es reconocida y se envía la petición al servidor.
Escenario	<ol style="list-style-type: none"> 1. Seleccionar modo manual de la aplicación. 2. La aplicación muestra por pantalla una lista de los elementos del sistema domótico. 3. El usuario selecciona un elemento del sistema domótico. 4. El sistema selecciona el elemento domótico e inicia la siguiente tarea. 5. El sistema inicia la búsqueda de las acciones posibles del elemento seleccionado. 6. El sistema muestra las acciones por pantalla. 7. El usuario selecciona de forma manual una acción a realizar. 8. El sistema realiza el mandato al servidor de la acción reconocida de forma automática.
Escenario Alternativo	--

Tabla 81.: CU-2.1 Mandato de elemento por selección táctil.

5. Descripción del sistema Domótico.

En este apartado, se van a describir todos los elementos del sistema domótico desarrollado para la integración de la aplicación móvil. Para ello, se iniciará con una breve introducción en la que se dará pie al sistema domótico, para seguidamente, describir los requisitos extraídos del sistema domótico.

Así, la integración con la aplicación, se podrá dar con cualquier sistema que cumpla los requisitos en dicho punto expuestos. Estos requisitos componen el sistema en sí. Para la descripción de la infraestructura del sistema, se puntualizará el servidor, el bus, y sus dispositivos.

Finalmente, se terminará con una breve descripción técnica de cada uno de los elementos necesarios para montar el sistema domótico.

Cabe añadir, que la implementación física de éste sistema domótico únicamente se ha podido realizar en el laboratorio de la empresa de domótica debido a los grandes costes de los dispositivos.

5.1 Introducción.

Dado que el sistema domótico se establecerá en el propio hogar de la persona de movilidad reducida, el sistema domótico deberá ser fiable, robusto y sencillo de utilizar.

A pesar de todo ello, como cualquier sistema domótico necesitará de un mantenimiento y de una cierta supervisión por parte de un administrador. Es por ello, que uno de los requisitos que se establecerán es el de la habilitación de un método de acceso remoto al servidor.

Se ha utilizado el estándar KNX para el desarrollo de este sistema^[5]. KNX es el estándar mundial para el control de casas y edificios. Aprobado como Estándar Internacional (ISO/IEC 14543-3), así como en el Estándar Europeo (CENELEC EN 50090 y CEN EN 13321-1) y Estándar en China (GB/Z 20965). Se garantiza que los productos KNX hechos por diferentes fabricantes pueden ser combinados. La propia marca registrada KNX garantiza la interoperabilidad y el "interworking".

Cabe destacar, que este estándar está basado en otros con más de 15 años de experiencia en el mercado. Entre otros, los sistemas predecesores de KNX son: EIB, EHS y BatiBUS.

Los medios de transmisión para este tipo de infraestructuras son de diversos tipos, van desde el par trenzado al IP/Ethernet. Sobre el medio de transmisión, se han de encontrar todos los dispositivos interconectados.

Los dispositivos conectados al bus, tanto sensores (en caso de necesitarlos), como actuadores, son utilizados para el control de equipamiento de gestión de edificios en todas las aplicaciones posibles: iluminación, persianas, sistemas de seguridad, gestión energética, calefacción, sistemas de ventilación y aire acondicionado, sistemas de supervisión y señalización, interfaces a servicios y sistemas de control de edificios etc.

Todas estas funciones pueden ser controladas y señalizadas utilizando un sistema uniforme sin la necesidad de centros de control adicionales.



Ilustración 17: Esquema que resume las funciones que se interconectarán.

5.2 Requisitos del sistema domótico.

Los requisitos de éste sistema domótico han sido expuestos en el apartado cuatro de requisitos del sistema. A pesar de ello, se van a exponer y justificar de forma textual en éste apartado.

Dado que el sistema ha de realizar una serie de acciones que le vienen en forma de petición 'HTTP request', el servidor, deberá ser capaz de realizar acciones de procesado, enrutado y respuesta. Con la tecnología actual en sistemas domóticos, el proceso de enrutado será desempeñado por un bus autónomo energéticamente. De éste modo, cuando le llegue un 'request' al servidor, éste, lo

procesará, identificará el dispositivo al que va dirigida dicha acción y finalmente mandará el mandato al dispositivo de la red a través del BUS de direccionamiento. Éste, es un punto importante, dado que el servidor ha de estar integrado con el BUS para que sea capaz de recibir y enviar la orden de la aplicación móvil.

No obstante, no termina ahí el trabajo del servidor. Éste, ha de procesar una respuesta del dispositivo (actuador en nuestro caso), diciéndole si el mandato enviado ha podido ser ejecutado exitosamente o no. Por último, dado que el servidor conoce si el mandato enviado al dispositivo ha tenido un resultado exitoso, se lo ha de comunicar a la aplicación móvil. Finalmente, la aplicación podrá informar del resultado de la petición realizada por el usuario.

Por lo tanto, se puede entender que el servidor ha de ser capaz de recibir peticiones 'HTTP request', enviar peticiones y además procesar los posibles mensajes de respuesta que le lleguen desde los dispositivos a través del BUS y comunicárselo a la aplicación móvil.

Dado que ya se ha realizado una explicación de los requisitos del servidor domótico y BUS de direccionamiento (a mayor carga del mismo), queda realizar una descripción de los requisitos que los dispositivos tienen.

Como el servidor domótico lanza una serie de mandatos sobre los mismos, éstos deben de estar preparados para las acciones posibles del elemento que controlan.

Por ejemplo, el dispositivo encargado de controlar la luz, tendrá que conocer sus acciones (encender o apagar la luz) y su estado (encendido o apagado). De este modo, cuando el servidor domótico le lance la petición de encender o apagar, éste estará listo y sabrá contestar éxito o no en función de que ésta haya sido realizada.

Para facilitar la comprensión del sistema que se desea obtener, que se expone un diagrama de alto nivel.

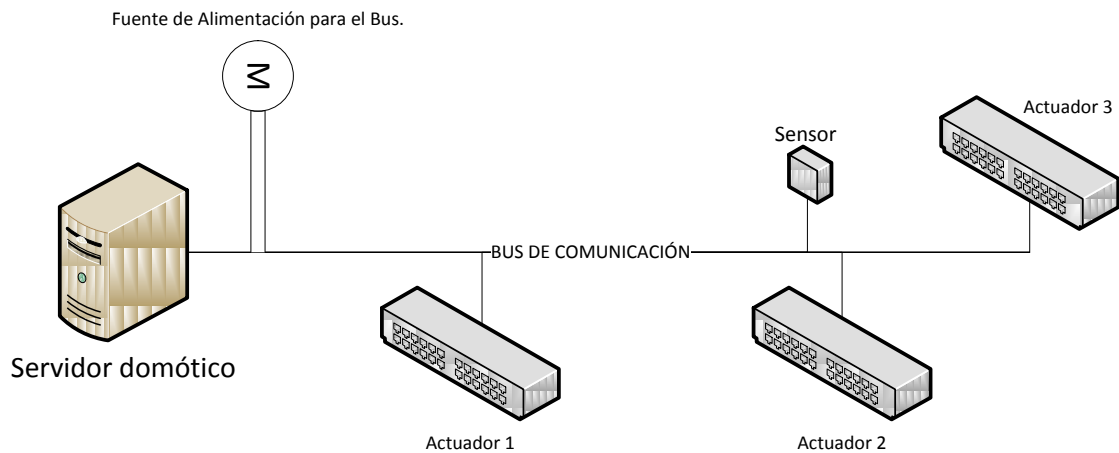


Ilustración 18: Esquema del sistema domótico asistencial.

5.3 Infraestructura del sistema.

El sistema domótico necesario para este proyecto, cuenta con los elementos básicos de un sistema de domótica tradicional.

Un servidor domótico (cliente IP) que realizará el trabajo de enviar los mandatos a los dispositivos de su red. Además, será necesario un BUS de datos que sea capaz de interconectar el servidor domótico (ó central IP) con los dispositivos que se encuentren domotizados.

Es decir, deberá ser capaz de enviar de un punto a otro las peticiones de los dispositivos. Además, deberá contar con los dispositivos que automatizan mediante un 'actuador' las acciones a realizar.

Por ejemplo, el servidor domótico conectado a una red, recibirá de la aplicación móvil un mandato por medio de un HTTP request. A partir de ese momento, el servidor, enviará dicha acción al dispositivo que corresponda, la acción a acometer por medio de un BUS de comunicación. Por último el dispositivo, deberá automatizar la acción recibida.

A continuación, se enumeran los dispositivos necesarios:

- Cliente IP.
- BUS de comunicaciones.
- Fuente de alimentación para el BUS de comunicaciones.
- Actuador y Sensor.
- Comunicador USB.

5.3.1 Servidor domótico.

El primer elemento que se ha querido especificar es el servidor domótico. Se trata de una central IP capaz de atender peticiones http request. El modelo que se ha seleccionado he implantado es el que se expone a continuación:

Dispositivo: Central IP

Objetivo: Atender peticiones 'HTTP'.

Código del producto: IPZ 1000 REG



Ilustración 19: Central IP de la instalación.

5.3.2 Fuente de alimentación (Bus).

El siguiente elemento que se ha querido especificar es la fuente de alimentación que dará corriente al BUS que unirá la central IP con el dispositivo de comunicación. El modelo elegido para ello, ha sido:

Dispositivo: Fuente de Alimentación (BUS).

Objetivo: Enrutar el tráfico entre la central IP y los módulos de comunicación de los dispositivos.

Código del producto: 2002 REG

Características: Cuenta con protección de sobrecargas y cortocircuitos del bus.



Ilustración 20: Fuente de Alimentación de la instalación.

5.3.3 Actuador.

El siguiente elemento que se ha querido especificar es el actuador, será el dispositivo encargado de mandar las acciones a realizar a los dispositivos. El modelo elegido para ello, ha sido:

Dispositivo: Actuador de 8 salidas.

Objetivo: Mandar las acciones a realizar al dispositivo final (Detector de movimiento, Detector de luz... etc).

Código del producto: 2308.16 REGGM

Características: Se trata de un actuador de 16 A (Amperios).

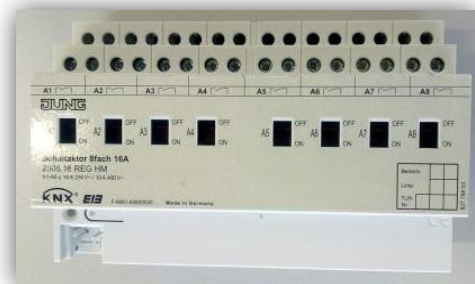


Ilustración 21: Actuador de la instalación.

5.3.4 Sensor.

El siguiente elemento que se ha querido especificar es el sensor, será el dispositivo encargado de conocer el estado del dispositivo. El modelo elegido para ello, ha sido:

Dispositivo: Sensor de luz.

Objetivo: Conocer el estado de la luz.

Código del producto: LS 3180-1

Características: Requiere de un acoplador de bus cod.:2070U.



Ilustración 22: Actuador de la instalación.

5.3.5 Módulo de comunicación USB.

El siguiente elemento que se ha querido especificar es el módulo de comunicación USB, será el dispositivo encargado de permitir la conexión de un detector o dispositivo y el ordenador o servidor domótico por medio de un puerto USB. El modelo elegido para ello, ha sido:

Dispositivo: Módulo de comunicación USB.

Objetivo: Mandar las acciones a realizar al dispositivo final (Detector de movimiento, Detector de luz... etc).

Código del producto: 2130 USB

Características: Contiene un puerto USB 2.0.



Ilustración 23: Módulo de comunicación USB de la instalación.

5.3.6 Otros componentes necesarios.

Finalmente, se exponen una serie de materiales algo triviales o no, para el montaje de la instalación. Como se puede observar en la ilustración, fueron necesarios terminales de conexión, una serie de Marcos para el montaje tanto vertical como horizontal y finalmente el cable del BUS:

Dispositivo: Materiales varios.

Objetivo: Completar la instalación.

Código del producto: Terminales de Conexión (2050RT SW), Marco para montaje (FD981WW), cable BUS J-Y

Características: Contiene un acoplador para el bus de código 2070U.



Ilustración 24: Otros componentes de la instalación.

6. Diseño del Sistema.

Una vez definida la gestión del proyecto con la metodología de desarrollo expuesta y con un análisis en profundidad realizado exponiendo todos los requisitos del sistema y sus posibles casos de uso, queda, diseñar el propio sistema.

En este apartado, se realizará una breve introducción del sistema a diseñar y posteriormente se expondrán los aspectos más importantes del mismo. Dado que ya se conocen todos los detalles del sistema, se puede apostar por un diseño que de antemano, se sabe que funcionará. Si el análisis ha sido lo suficientemente eficaz, seremos capaces de realizar un diseño eficiente del sistema.

6.1 Introducción.

Una vez definidos todos los requisitos del sistema, se puede iniciar el proceso de diseño del sistema. Para ello, se iniciará un estudio de las opciones de diseño para el proyecto. Seguidamente, se expondrá en un breve diagrama la arquitectura del sistema.

6.2 Opciones de Diseño.

A pesar de que el proyecto que se estudia tiene limitados lenguajes de programación y de base de datos, se considera de gran importancia el estudio y justificación de los elementos empleados.

Es por ello, que en primer lugar, se expondrán las opciones de lenguajes de programación, seguidamente, las alternativas existentes en base de datos y para terminar, se expondrán las opciones en cuanto a interfaz se refiere y servicios se refiere.

6.2.1 Lenguajes de Programación.

El lenguaje de programación nativo en Android es Java, como primera opción se seleccionó este lenguaje como el ideal.

No obstante, se estudió la posibilidad del desarrollo con HTML5 como lenguaje de programación. Éste último lenguaje, se considera como uno de los nuevos modos para desarrollar aplicaciones móviles adaptables a todos los

sistemas operativos móviles del mercado. Éstas, fueron las razones que llevaron al estudio de dicha posibilidad.

El sistema a desarrollar requiere de un acceso de muy bajo nivel a las librerías de Android, además, como se verá más adelante, requiere de servicios específicos de Google, como el 'Google Translate'. En la siguiente tabla se realiza una comparación entre los aspectos más importantes de un lenguaje de programación para el proyecto.

Lenguaje de Programación Aspecto a destacar	HTML5 	JAVA 
Librerías de Android	Alta	Muy Alta
Traducción de texto.	Media	Muy Alta
Reconocimiento por Voz.	Media	Muy Alta
Lenguaje de BBDD-	Web SQL Database	SQLite
Reproducción de Texto sonoro.	Muy Baja	Muy Alta
Peticiones a Servidor.	Media	Alta
Extracción del lenguaje del dispositivo.	Alta pero costosa.	Muy Alta

Tabla 82.: Comparación entre HTML5 y Java.

Como se puede observar en la tabla anterior, la integración de Java con los sistemas Android como la base de datos nativa (SQLite) o Google Voice Recognizer, son totales. Mientras que HTML5 usa su lenguaje de base de datos y el módulo de reconocimiento por voz es menos fiable que el integrado por Google.

Otro aspecto importante, es el servicio de reproducción de texto de forma sonora. En Android, se trata de un servicio impecable (TextToSpeech) en HTML5 el servicio es eficaz pero de mayor dificultad en su implantación. Por último las peticiones a servidor desde HTML5 no están muy integradas de por sí, y se debería aplicar una solución externa, mientras que en Java se encuentran integradas.

Finalmente, se opta por la solución inicial, se recurrirá al lenguaje nativo del dispositivo para poder tener un mayor dominio del mismo.

6.2.2 Lenguajes de Bases de Datos.

En el apartado anterior, se ha elegido el lenguaje de programación nativo como lenguaje para el proyecto, debido ello, la única posibilidad de programación en base de datos, es la de SQLite. Por ello, no se abarcó ningún estudio al respecto de un lenguaje distinto.

6.2.3 Interfaz y Servicios.

Finalmente, quedó el estudio de las interfaces y servicios a requerir. Para ello, en primer lugar, se valorarán las interfaces y número de interfaces que se han de desarrollar para el proyecto.

- Interfaces:

De cara a este estudio, se expusieron distintas interfaces para el sistema y flujo de las mismas. No se describirán todas las opciones diseñadas pero se justificará la elección de las interfaces finales. Se proponen las siguientes interfaces:

- Primera Interfaz → Única interfaz parametrizable y de obligado uso táctil. En el caso de una persona impedida será la única ayuda que requerirá para el uso de la aplicación. Se definen dos opciones, la primera 'Manual Mode', la segunda 'Automatic Mode'. Por último se define un botón 'GO!' para iniciar el modo.

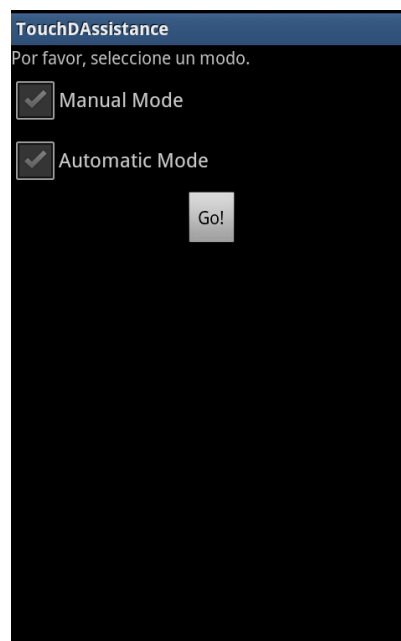


Ilustración 25: Primera interfaz del sistema.

- Segunda Interfaz → Breves instrucciones textuales (opcional) y una lista con todos los elementos del sistema domótico. Botón 'Back' para una posible marcha atrás (modo manual).

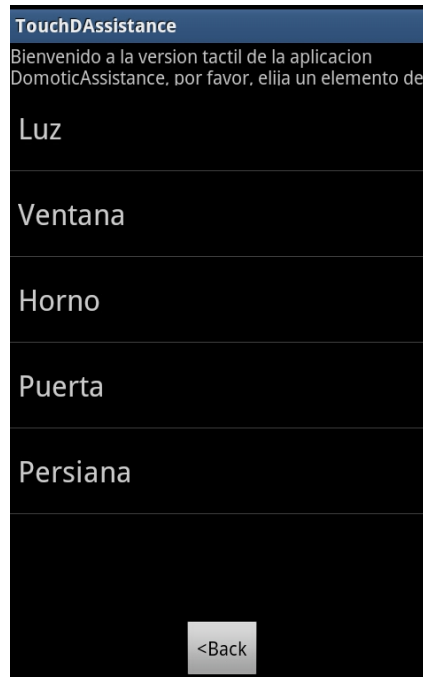


Ilustración 26: Segunda interfaz del sistema.

- Tercera Interfaz → Breves instrucciones textuales (opcional) y una lista con todos los elementos del sistema domótico. Botón 'Back' para una posible marcha atrás (modo manual).

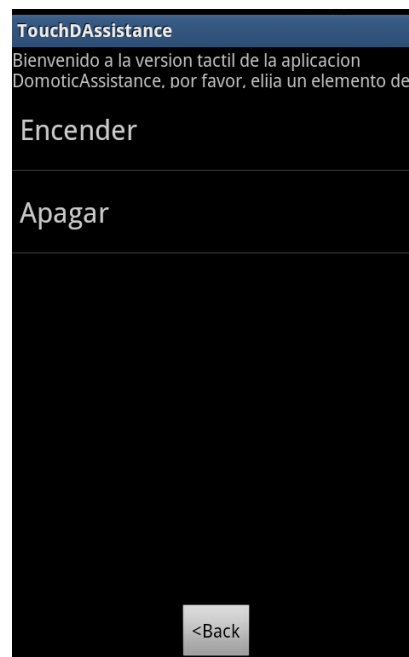


Ilustración 27: Tercera interfaz del sistema.

Se ha de apuntar que se definen las mismas interfaces de usuario para cada uno de los modos. No obstante, se controlarán por objetos Java distintos y su comportamiento también lo será.

- Servicios:

De cara a seleccionar los servicios adecuados para la aplicación, surgen ciertas dudas en cuanto a servicios concretos.

El servicio de “TextToSpeech” para la reproducción de texto en forma sonora, resulta evidente, así como el servicio de reconocimiento de voz de Google, “Google Voice Recognizer”. No obstante, el servicio de “Google Translate” como sistema de traducción, no está tan claro.

En el año 2010, Google privatizó el uso de su API (librería) de traducción, declarando el uso de “Google Translate” como un servicio de pago. Se pretende una austeridad completa en los recursos necesarios para el proyecto, por ello, se estudian distintas librerías de traducción como “Bing”, o “Yahoo”. No obstante, todas ellas, de nuevo, resultan soluciones de pago.

A partir de estas conclusiones, y a falta de servicios de traducción gratuitos, se estudió la mejor opción según el servicio y su precio.



Servicio de Traducción Costes según # de Caracteres	Bing Microsoft Translator 	Google Translate 
1 Mill de char/mes	0,00\$	20,00\$
4 Mill de char/mes	40,00\$	80,00\$
6 Mill de char/mes	60,00\$	120,00\$
8 Mill de char/mes	80,00\$	160,00\$

Tabla 83.: Comparación de servicios de traducción.

Tras una serie de cálculos de uso de la aplicación, se encontró que Google Translate es el mejor servicio en relación calidad precio. A pesar de tener un coste económico mayor (prácticamente el doble) y a pesar de que ‘Bing’ tiene una versión gratuita de hasta el millón de caracteres, el servicio, no diferencia entre el uso de 2,3 Mill caracteres/mes y 3,9 Mill caracteres/mes y

sin embargo, Google, sí lo hace. Es decir, la facturación siempre será más ajustada a nuestro uso en Google y no así en Microsoft. No obstante, debido a su reducido coste, no

Como se puede observar en la tabla anterior, en el caso de “Google Translate”, el primer millón de caracteres traducidos paga la cuota mínima de 20\$. Dado que la aplicación no va a tener un flujo de traducción excesivo, se considera como una buena opción.

Finalmente, se desechó la opción de utilizar el servicio de Microsoft Bing debido al apoyo inexistente a la plataforma Android. Bing, considera ‘Windows Phone’ como el sistema operativo desde el que se ha de consultar su API (librería).

6.3 Arquitectura del Sistema.

Una vez elegidos todos los aspectos técnicos relevantes del sistema se pasa a exponer un diagrama de la arquitectura del sistema final. A continuación se expone en una ilustración:

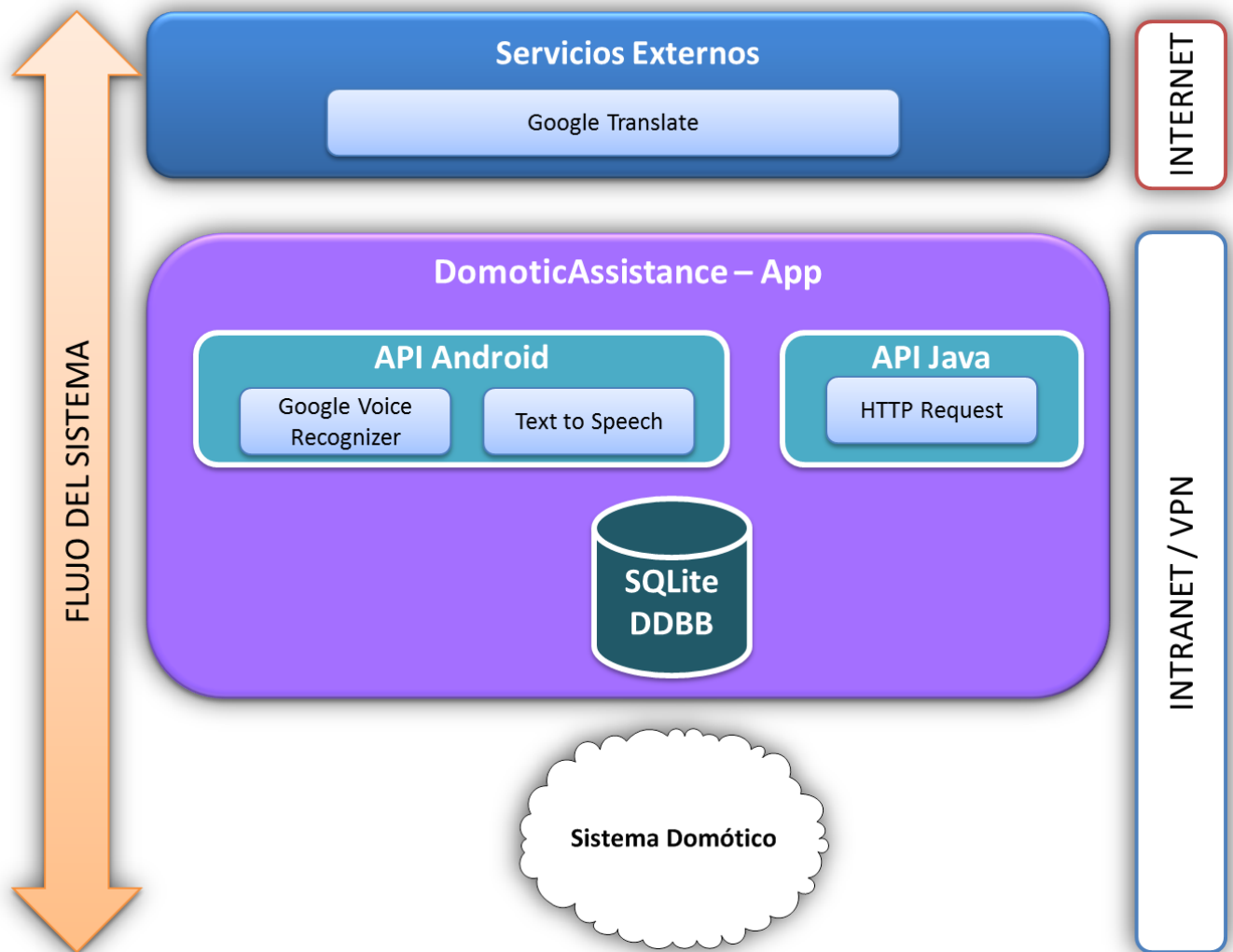


Ilustración 28: Arquitectura del sistema.

6.4 Estructura de la Base de Datos.

La estructura de Base de Datos requerida para el sistema es la descrita en el siguiente diagrama de bases de datos.

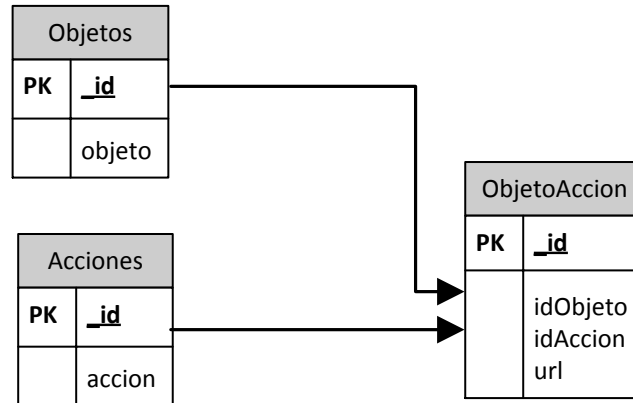


Ilustración 29: Estructura de base de datos del sistema.

6.5 Diseño de UML del sistema.

Dado que se trata de un proyecto Java a pesar de que se acceda a las librerías de Android, se ha creído conveniente la exposición del diseño UML del sistema. A continuación, se expone la ilustración con la descripción completa de las actividades (clases Java) del sistema:

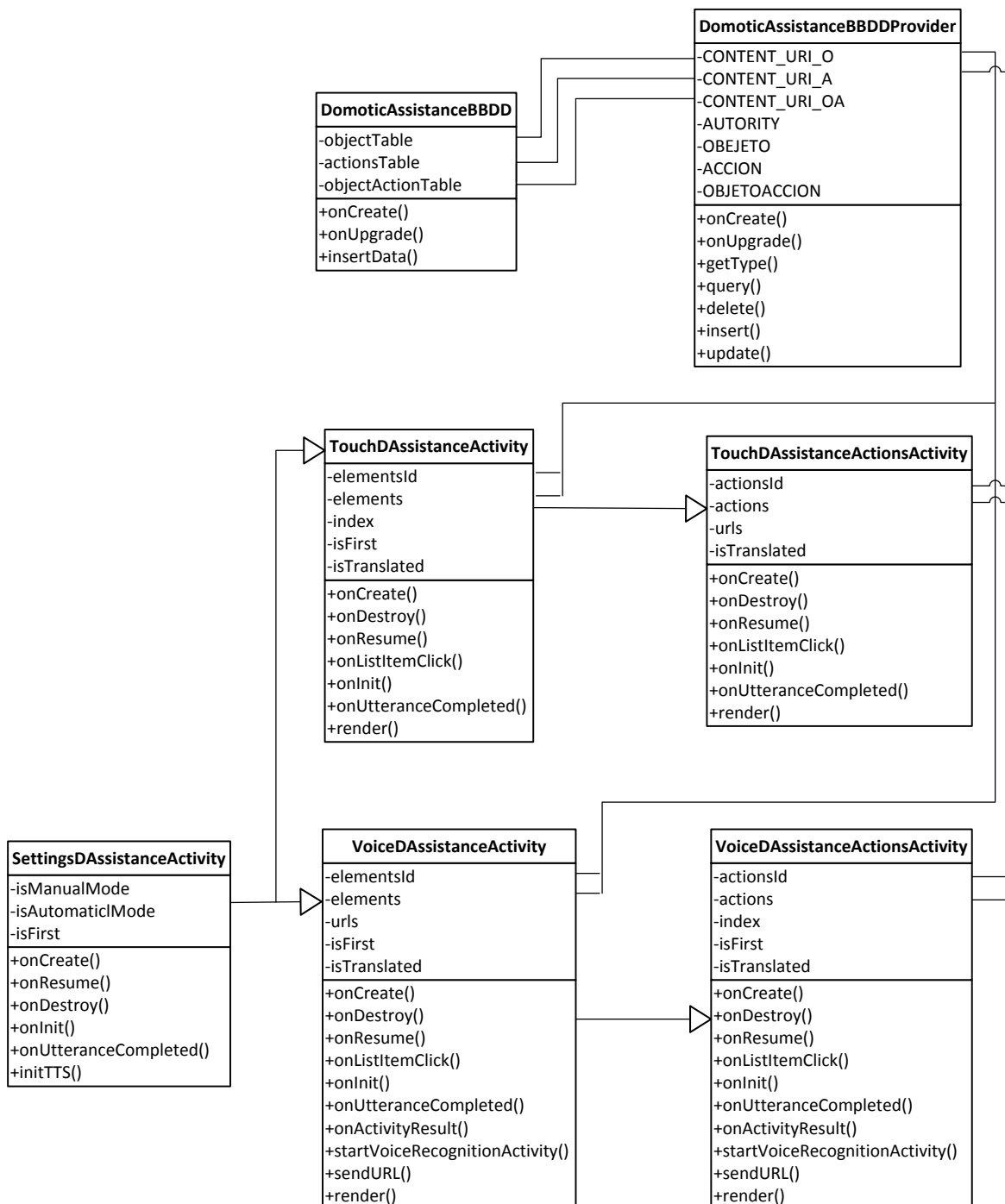


Ilustración 30: Sistema UML del sistema.

6.6 Interfaces necesarios para el sistema.

Una vez definida la arquitectura del sistema, el diseño UML de la aplicación y el modelo relacional de la base de datos, se pasa a definir las interfaces necesarias para el sistema.

Dado que los casos de uso de la aplicación ya se encuentran definidos, se pueden diseñar las interfaces de usuario necesarias para el sistema. A continuación se pasa a mostrar el flujo de las interfaces de usuario de los dos modos posibles de la aplicación:

En primer lugar, se muestra el flujo de interfaces en el modo automático con reconocimiento de voz.

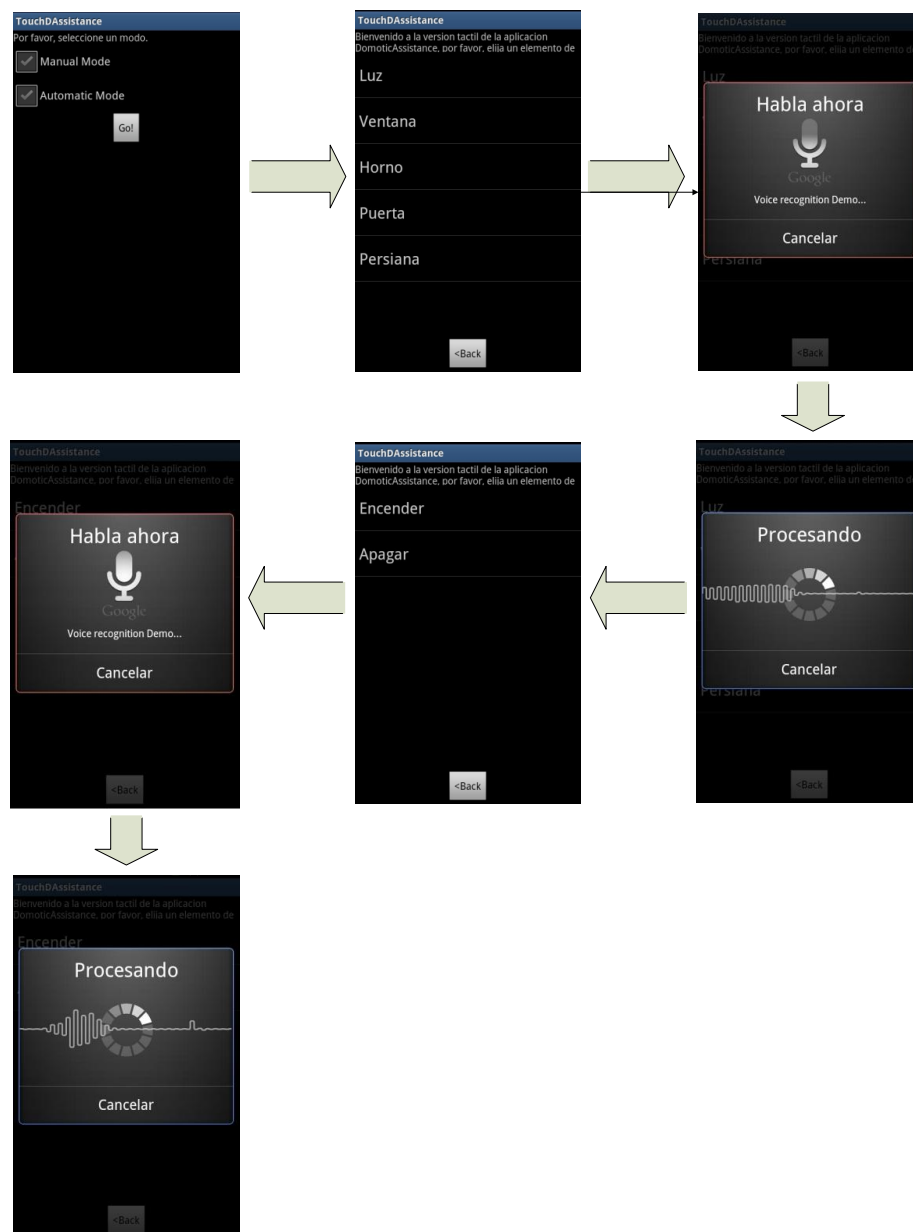


Ilustración 31: Interfaces de usuario y su flujo del modo automático.

Finalmente, el modo manual, se trata de un modo más sencillo, puesto que está controlado de forma táctil.



Ilustración 32: Interfaces de usuario y su flujo del modo manual.

7. Implementación del Sistema.

Este apartado, tratará de aclarar al lector los métodos utilizados por el equipo de desarrollo para poder llevar a cabo el proyecto. Para ello, se especificarán las herramientas necesarias para el desarrollo de éste sistema, así como el entorno tecnológico completo utilizado.

Sin entrar a comentar ninguna línea del código de la aplicación, se tratará de realizar una breve descripción de cada una de las partes implementadas. Además, se tratará de aclarar cualquier duda respecto al desarrollo del sistema.

7.1 Introducción.

Como se ha apuntado anteriormente este apartado describirá la forma de implementar el proyecto planteado. Declarando qué elementos se han utilizado para el desarrollo del proyecto y justificando su uso, se podrá comprobar que los recursos materiales especificados en el apartado tres, de gestión del proyecto, han sido requeridos para el proyecto, y porqué.

Es por ello, que se iniciará por una breve descripción de todas las herramientas de desarrollo del sistema y a continuación, se especificarán los servicios de los que han tenido que hacer uso el proyecto.

7.2 Herramientas de desarrollo del sistema.

Las herramientas utilizadas para el desarrollo del sistema han sido la gran mayoría de 'open source'. Es decir, libres para su uso sin necesidad de realizar ningún pago para obtener las licencias.

7.2.1 Eclipse.

Se ha tenido que utilizar la distribución de Eclipse "Juno" la cual es la más reciente en la actualidad. Como se ha apuntado anteriormente, Eclipse es un framework de trabajo de libre disposición y no se ha tenido que pagar nada. El recurso está disponible en la siguiente URL:

<http://www.eclipse.org/downloads/>

No obstante, para el desarrollo de aplicaciones en Android, se ha de instalar un plug-in necesario para ello. Google principalmente provee dos plug-in's

para desarrolladores. El primero, es el correspondiente al desarrollo de aplicaciones en App – Engine. El segundo, es el correspondiente con el desarrollo de aplicaciones en Android, el cual es el que nos interesa. Dicho plug-in, se ha de instalar del modo que se detalla en la siguiente URL oficial de Google.

<https://developers.google.com/eclipse/docs/install-eclipse-4.2>

7.2.2 SDK Android.

Por otro lado, el desarrollador que vaya a implementar la aplicación Android, tendrá que descargarse el SDK de Android. De este modo, se ha enlazar la configuración del plug-in de Eclipse. La descarga estará disponible en la siguiente URL:

<http://developer.android.com/sdk/index.html>

Esta herramienta es necesaria para poder realizar una buena integración del marco de desarrollo. Con Eclipse y el SDK de Android integrados, se puede incluso realizar pruebas sobre dispositivos Android emulados. Por tanto, se puede asumir que un escenario mínimo para el desarrollo de éste tipo de aplicaciones sería la integración entre ambos.

7.2.3 SQL Admin.

En el caso del proyecto actual, el equipo de desarrollo, trabajará con la base de datos instalada en los dispositivos móviles de Android. Éste es el denominado lenguaje “SQLite” el cual es muy similar a MySQL como lenguaje de base de datos, pero no igual.

Los administradores de BBDD de MySQL no sirven para los de éste lenguaje, y lo más acertado es la descarga de un administrador de éste tipo de bases de datos. Se ha encontrado un administrador de SQLite que es de código abierto, y goza de gran reputación entre los desarrolladores, éste es SQL Admin.

Éste software, únicamente nos servirá para poder comprobar de forma explícita los datos introducidos en la base de datos del dispositivo. De tal modo que, descargando el archivo ‘.sql’ del dispositivo, éste, pueda ser accedido por medio del administrador ‘SQL Admin’.

Dicho administrador, deberá estar disponible para la descarga gratuita en el siguiente enlace:

<http://sqliteadmin.orbmu2k.de/>

7.2.4 Microsoft Office 2010.

Finalmente, y debido a la gran carga de trabajo tanto de redacción como de maquetación, se accedió a la compra de una licencia de Microsoft Office. De este modo, la presente memoria está redactada sobre el mismo.

Debido a que es un recurso de pago, no se encuentra disponible para su descarga.

7.3 Servicios necesarios para el sistema.

En este apartado se describirán los servicios de los que requerirá la aplicación para realizar todas las funcionalidades necesarias. Para ello, se realizará una breve exposición del servicio y su uso en el sistema.

7.3.1 Google Voice Recognizer:

Se trata del servicio de reconocimiento de voz nativo en Android. Éste servicio será utilizado en cuatro ocasiones durante el ‘modo automático’ de la aplicación. Como se puede observar en el diagrama final, el servicio es utilizado para reconocer la voz del usuario y seguidamente para procesar la voz y así poder reconocer los elementos y las acciones que se desean activar. El logo en el diagrama de servicios para representar “Google Voice Recognizer”, será el siguiente:



Ilustración 33: Uso de Google Voice Recognizer.

Éste servicio se encuentra integrado en el sistema operativo ‘Android’. Es accesible desde la propia API de Android. Éste servicio realiza un reconocimiento por voz que ofrece resultados muy fiables.

Se trata de un servicio gratuito, no obstante se requiere de conexión a Internet para que se produzca el reconocimiento de la frecuencia de voz y por tanto del mandato vocalizado. Cabe señalar que el sistema ha dado tiempos de ejecución aceptable en la mayor parte de las pruebas.

7.3.2 Text To Speech:

Éste es el sistema de reproducción de texto en forma sonora nativo en Android. Éste servicio será usado de forma muy frecuente en la aplicación para mejorar la usabilidad de la misma. Se expone la ilustración final del diagrama de servicios en la que se describen los momentos en los que éste, es activado. Se puede observar que es el servicio más utilizado en el sistema. Éste será el logo utilizado para representar en el diagrama:



Ilustración 34: Uso de Google Text To Speech

De nuevo, se trata de un servicio integrado en el propio sistema operativo Android. Es accesible desde la propia API de Android y se trata de un servicio parametrizable. Éste servicio reproduce en voz alta un texto previamente introducido.

Una vez más, se trata de un servicio gratuito y fácilmente integrable en el sistema. Éste servicio puede ser ajustable de forma automática. En el sistema actual se pretende implantar con acciones en todos los idiomas disponibles en Android.

7.3.3 Google Translate^[6]:

El servicio de Google externo al sistema operativo Android que permite la traducción entre más de 62 lenguajes. Para el caso del proyecto actual, únicamente interesan los 33 lenguajes de los que dispone Android. Éste servicio se utilizará en tres momentos de la aplicación. A continuación se muestra el logo con el que será representado el servicio en el diagrama final:



Ilustración 35: Logo de Google Translate

El servicio de mayor coste económico es el de traducción del lenguaje. Se escoge el servicio de 'Google Translate' debido a su fiabilidad contrastada y su precio. Dado que no se encontraron librerías de traducción gratuitas, se ha optado por la elección de este servicio.

Se puede integrar fácilmente en Android de tal modo que queden traducidas las aplicaciones y su contenido de forma completa. No obstante, conviene señalar que en futuras implementaciones se podrían buscar alternativas gratuitas.

Para la implementación de éste servicio no se requiere de una gran computación en el dispositivo, no obstante, se requiere de conectividad a internet.

7.3.4 HTTP Request [7]:

Finalmente, este servicio se encuentra integrado en la API de Java. Este servicio es utilizado cada vez que se quiere enviar un mandato al servidor. A continuación se muestra el logo representativo del momento en el que se utiliza el servicio:

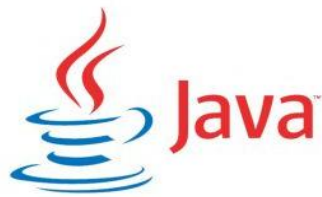


Ilustración 36: Logo de Java, API utilizada para el servicio HTTP Request.

El último servicio del cual requiere la aplicación es de un servicio elemental en la librería del protocolo 'HTTP' en Java. Ésta librería permite interactuar con el protocolo 'HTTP' de forma íntegra y muy sencilla.

No obstante, cabe señalar que dado que las operaciones de las que requiere nuestra aplicación son muy sencillas, un servicio menos extenso que el que nos provee Java es también adaptable.

A continuación se muestra el diagrama final del uso de servicios. Se han utilizado las interfaces de usuario para re los momentos en los que los servicios pueden ser activados.

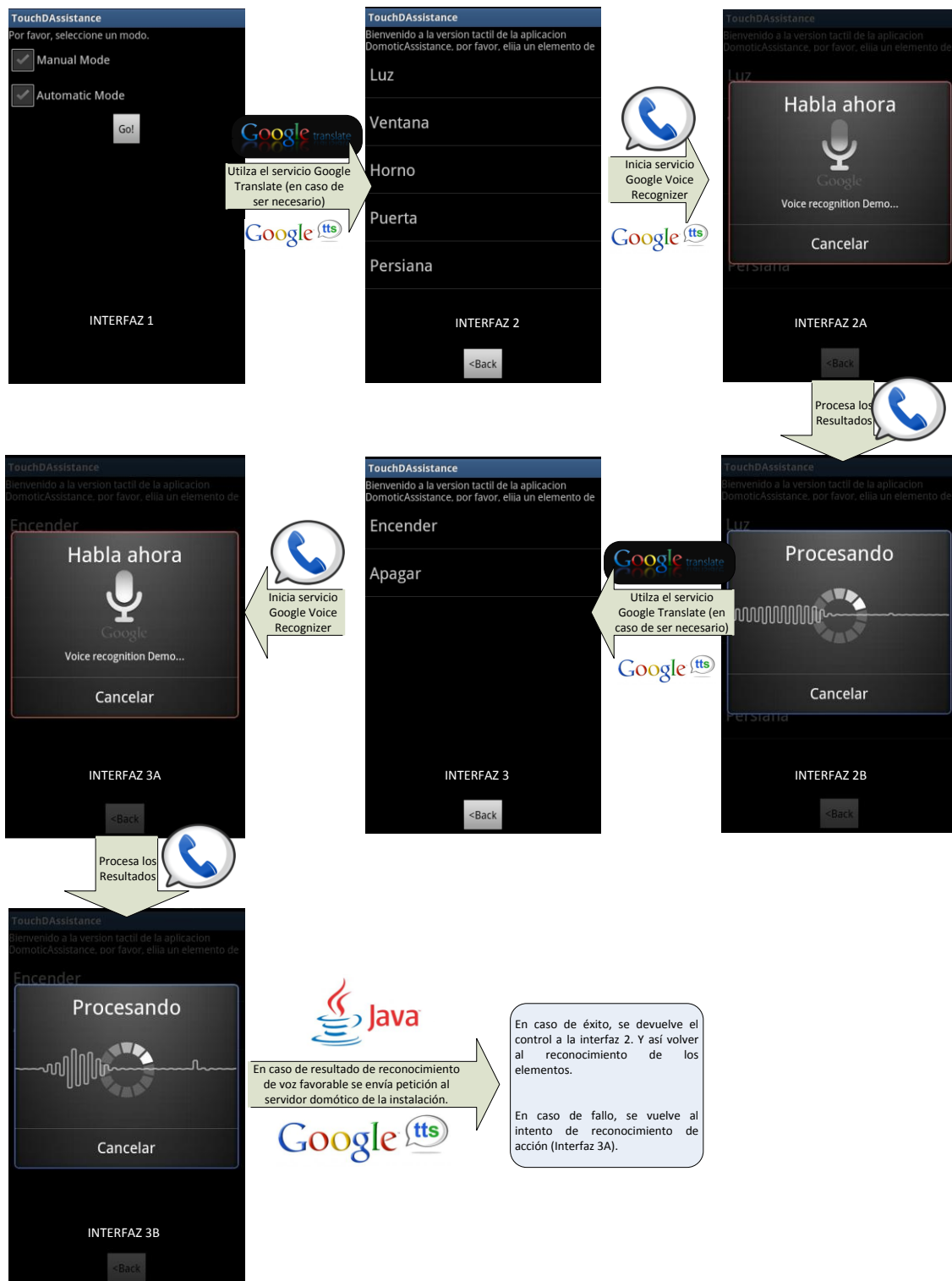


Ilustración 37: Diagrama de uso de servicios.

7.4 Sistema Android.

El módulo de reconocimiento de voz para domótica asistencial divide sus desarrollos principalmente, en dos campos. El primero, el sistema de archivos Java, a pesar de no explicar el código en profundidad, se expondrán ciertos rasgos característicos de cada clase.

La segunda parte, es la de la administración de los datos por medio de una base de datos. Para ello, se ha empleado SQLite y en este apartado, se expondrán de forma breve las consultas y accesos necesarios implementados en la aplicación.

7.4.1 Clases Java del sistema.

El sistema de archivos de la aplicación, se divide en cuatro paquetes. Como se puede observar en la ilustración, todos los paquetes siguen una nomenclatura constante 'com.android.touchDAssistance', a partir de ahí el nombre determina su función dentro de la aplicación:

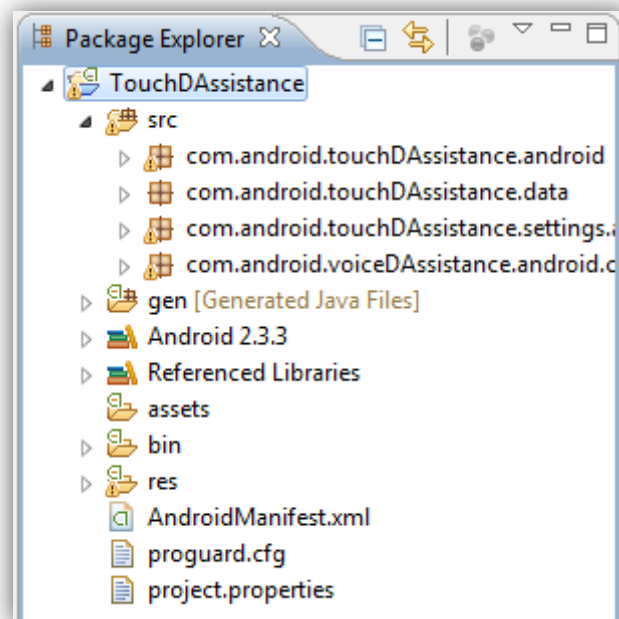


Ilustración 38: Sistema de ficheros de la aplicación.

El primero de los paquetes, el paquete 'android'. En él, se encuentran las Actividades encargadas de iniciar las actividades del modo manual de la aplicación. Además llevan toda la lógica de dicho modo.

Seguidamente se encuentra el paquete 'data', éste, es el paquete en el que se encuentra el proveedor de datos y la actividad encargada de automatizar la primera carga de datos al dispositivo.

A continuación, se puede ver el paquete 'settings', éste paquete contiene la Actividad encargada de iniciar la aplicación y mostrar la interfaz principal. Ésta interfaz será en la que se seleccione el modo de la aplicación.

Por último, se puede observar el único paquete que cambia la nomenclatura general de los mismos, éste es el 'com.android.voiceDAssistance.android'. Éste paquete es el que contiene las dos actividades que inician cada una de las interfaces del modo automático de la aplicación. Éstas, contienen toda la lógica de la aplicación de éste modo.

7.4.2 Base de Datos en SQLite.

En este apartado se especificarán las consultas que se ejecutan sobre la base de datos con un lenguaje sin tecnicismos.

En primer lugar, en los dos modos de la aplicación se ejecuta una consulta de la base de datos en el momento en el que se muestra la primera interfaz exclusiva del modo. Para ello, se consulta sobre la base de datos todos los elementos disponibles en el sistema domótico.

La siguiente consulta que se ejecuta sobre la base de datos se produce en el momento de mostrar las acciones disponibles para el elemento seleccionado.

En ambos modos este caso se produce de igual forma, el sistema ya ha identificado el elemento seleccionado, por lo que en primera instancia, se consultan las posibles acciones del elemento. A la vez que se obtienen las acciones del elemento seleccionado, se obtiene la URL a mandar al servidor.

Éstas son las consultas que se ejecutan sobre la base de datos para obtener las funcionalidades del sistema.

8. Pruebas del Sistema.

En este apartado se especificarán las pruebas realizadas sobre el sistema. Se realizará una primera introducción en la que se especifique la forma de proceder para desarrollar un plan de pruebas efectivo. En dicha introducción se tratará de dar pie a la explicación de los sucesivos puntos del apartado así como exponer los casos de prueba.

8.1 Introducción.

Para especificar un plan de pruebas válido y consistente, en primer lugar se expondrán los procedimientos para desarrollar las pruebas. Se destacan tres tipos de prueba. La primera, de infraestructura hardware, la segunda, por flujo de aplicación y finalmente por funcionalidad de la aplicación.

8.2 Procedimientos de pruebas.

Se van a realizar distintos tipos de pruebas para obtener una validación total del sistema. En primer lugar, se especificarán las pruebas de hardware, estas, constarán de pruebas del dispositivo de cara al sistema. Dado que se trata de una aplicación móvil, las pruebas se perfilan más hacia el lado de la usabilidad. Se deja así un plano más técnico de pruebas al apartado de las pruebas de funcionalidad.

Seguidamente, se expondrán las pruebas de flujo de la aplicación, en ellas, se probarán el correcto flujo de las distintas interfaces de la aplicación. Estas pruebas se consideran de vital importancia dado que se debe asegurar un modo automático de la aplicación adaptado para personas con discapacidad.

Finalmente, se presentarán las pruebas de funcionalidad de la aplicación. Estas pruebas, serán 'debugeadas' mediante el framework de Eclipse para verificar el perfecto funcionamiento de cada servicio del que hace uso la aplicación.

Para finalizar este apartado, se realizará una breve explicación de todos los campos utilizados en la plantilla para describir las pruebas del sistema.

- **Identificador:** Identifica de forma unívoca cada uno de las pruebas del sistema que se explicarán a continuación. El formato para las pruebas del sistema será **PXXX-YY**, siendo **XXX** el tipo de prueba y **YY** la numeración de la misma.
- **Descripción:** Breve descripción de la prueba a realizar.
- **Requisito de Software:** Relaciona la prueba con un requisito de software ya declarado.

- **Estado Inicial:** Se describen los pre-requisitos del sistema para la prueba.
- **Estado Final:** Se describe la salida o producto de la prueba.
- **Procedimiento de la prueba:** Se realiza una breve descripción del procedimiento a realizar para reproducir la prueba.
- **Resultado de la Prueba:** Se expone el resultado de la prueba realizada. Valores posibles: 'Éxito' o 'Fallo'.

Identificador	PXXX – YY
Descripción	
Requisito Software	
Estado inicial	
Estado Final	
Procedimiento de Prueba	
Resultado de la Prueba	

Tabla 84.: Plantilla de Pruebas del sistema.

8.2.1 Pruebas de infraestructura hardware.

Estas pruebas se definirán en el identificador como 'PHAR – YY'. Se centrarán en probar el funcionamiento del hardware y su integración en la aplicación.

Identificador	PHAR – 01
Descripción	Prueba de Sistema Operativo
Requisito Software	RSF – 01, RSF – 02, RSF – 19, RSF – 20
Estado inicial	Dispositivo móvil apagado.
Estado Final	Dispositivo móvil encendido.
Procedimiento de Prueba	Se ha de pulsar el botón de encendido hasta el inicio del dispositivo. Ingresar en ingresar en los ajustes del sistema. Se ha de pulsar la opción 'Acerca del teléfono'. Se ha de identificar en la etiqueta 'Versión de Android' al menos la versión: 2.3.3
Resultado de la Prueba	Éxito

Tabla 85.: Prueba PHAR – 01.

Identificador	PHAR – 02
Descripción	Prueba del funcionamiento de micrófono
Requisito Software	RSF – 03
Estado inicial	Dispositivo encendido.
Estado Final	Dispositivo reconoce voz .
Procedimiento de Prueba	Iniciar una llamada de voz. En caso de poder comunicarse, se verifica el funcionamiento del micrófono.
Resultado de la Prueba	Éxito

Tabla 86.: Prueba PHAR – 02.

Identificador	PHAR – 03
Descripción	Prueba del funcionamiento de cámara
Requisito Software	RSF – 03, RSF – 04, RSF – 21, RSF – 22.
Estado inicial	Dispositivo encendido.
Estado Final	Dispositivo captura una imagen.
Procedimiento de Prueba	Pulsar el botón de cámara del dispositivo. Pulsar el botón de ‘hacer foto’ de la cámara. Verificar la foto realizada.
Resultado de la Prueba	Éxito

Tabla 87.: Prueba PHAR – 03.

Identificador	PHAR – 04
Descripción	Prueba del funcionamiento del altavoz
Requisito Software	RSF – 03, RSF – 26, RSF – 27
Estado inicial	Dispositivo encendido.
Estado Final	Dispositivo reconoce voz y reproduce sonido.
Procedimiento de Prueba	Pulsar el botón físico del dispositivo de ‘Subir volumen’. Se ha de escuchar al final un tono indicando el volumen del terminal.
Resultado de la Prueba	Éxito

Tabla 88.: Prueba PHAR – 04.

8.2.2 Pruebas de flujo de la aplicación.

Estas pruebas se definirán en el identificador como 'PFLU – YY'. Se centrarán en probar el funcionamiento de las interfaces de la aplicación y su correcto flujo.

Identificador	PFLU – 01
Descripción	Prueba del flujo de interfaz, de pantalla inicial a pantalla de elementos (modo manual)
Requisito Software	RSF – 32, RSF – 18
Estado inicial	Aplicación instalada en el dispositivo.
Estado Final	Aplicación muestra interfaz de elementos del sistema.
Procedimiento de Prueba	Iniciar la aplicación. Chequear el modo manual de la aplicación. Pulsar botón 'Go!' de la interfaz.
Resultado de la Prueba	Éxito

Tabla 89.: Prueba PFLU – 01.

Identificador	PFLU – 02
Descripción	Prueba del flujo de interfaz, de pantalla de elementos a pantalla de acciones (modo manual)
Requisito Software	RSF – 06, RSF – 17, RSF – 32
Estado inicial	Aplicación mostrando interfaz de elementos.
Estado Final	Aplicación muestra interfaz de acciones del elemento.
Procedimiento de Prueba	Pulsar elemento de la interfaz.
Resultado de la Prueba	Éxito

Tabla 90.: Prueba PFLU – 02.

Identificador	PFLU – 03
Descripción	Prueba del flujo de interfaz, de pantalla de acciones a pantalla de elementos (modo manual)
Requisito Software	RSF – 32, RSF – 18
Estado inicial	Aplicación mostrando interfaz de acciones.
Estado Final	Aplicación muestra interfaz de elementos del sistema.
Procedimiento de Prueba	Pulsar botón 'Back' de la interfaz.
Resultado de la Prueba	Éxito

Tabla 91.: Prueba PFLU – 03.

Identificador	PFLU – 04
Descripción	Prueba del flujo de interfaz, de pantalla de elementos a pantalla de ajustes (modo manual)
Requisito Software	RSF – 32
Estado inicial	Aplicación mostrando interfaz de elementos.
Estado Final	Aplicación muestra interfaz de ajustes del sistema.
Procedimiento de Prueba	Pulsar botón 'Back' de la interfaz.
Resultado de la Prueba	Éxito

Tabla 92.: Prueba PFLU – 04.

Identificador	PFLU – 05
Descripción	Prueba del flujo de interfaz, de pantalla inicial a pantalla de elementos (modo automático)
Requisito Software	RSF – 33
Estado inicial	Aplicación instalada en el dispositivo.
Estado Final	Aplicación muestra interfaz de elementos del sistema.
Procedimiento de Prueba	Iniciar la aplicación. Chekear el modo automático de la aplicación. Pulsar botón 'Go!' de la interfaz.
Resultado de la Prueba	Éxito

Tabla 93.: Prueba PFLU – 05.

Identificador	PFLU – 06
Descripción	Prueba del flujo de interfaz, de pantalla de elementos a pantalla de acciones (modo automático)
Requisito Software	RSF – 05, RSF – 23, RSF – 33
Estado inicial	Aplicación mostrando interfaz de elementos.
Estado Final	Aplicación muestra interfaz de acciones del elemento.
Procedimiento de Prueba	Decir de forma vocal, el elemento de la interfaz.
Resultado de la Prueba	Éxito

Tabla 94.: Prueba PFLU – 06.

Identificador	PFLU – 07
Descripción	Prueba del flujo de interfaz, de pantalla de acciones a pantalla de elementos (modo automático)
Requisito Software	RSF – 07 , RSF – 33
Estado inicial	Aplicación mostrando interfaz de acciones.
Estado Final	Aplicación muestra interfaz de elementos del sistema.
Procedimiento de Prueba	Decir de forma vocal, la acción del elemento seleccionado anteriormente.
Resultado de la Prueba	Éxito

Tabla 95.: Prueba PFLU – 07.

Identificador	PFLU – 08
Descripción	Prueba del flujo de interfaz, de pantalla de reconocimiento de voz fallido a pantalla de reconocimiento de voz (modo automático)
Requisito Software	RSF – 07 , RSF – 33, RSF – 24, RSF – 25.
Estado inicial	Aplicación mostrando interfaz de acciones.
Estado Final	Aplicación muestra interfaz de elementos del sistema.
Procedimiento de Prueba	Decir erróneamente, de forma vocal, la acción del elemento seleccionado anteriormente. El sistema emitirá de forma sonora un mensaje textual de error. El sistema deberá volver a mostrar la interfaz de reconocimiento de voz.
Resultado de la Prueba	Éxito

Tabla 96.: Prueba PFLU – 08.

8.2.3 Pruebas de funcionalidad.

Estas pruebas se definirán en el identificador como 'PFUN – YY'. Se centrarán en probar el funcionamiento de las interfaces de la aplicación y su correcto flujo.

Identificador	PFUN – 01
Descripción	Prueba del servicio Text To Speech.
Requisito Software	RSF – 31
Estado inicial	Aplicación instalada en el dispositivo.
Estado Final	Aplicación en cualquiera de los dos modos de la aplicación mostrando interfaz de elementos.
Procedimiento de Prueba	<p>Iniciar la aplicación.</p> <p>Chekear la opción de Manual.</p> <p>Pulsar el botón 'Go!' de la interfaz.</p> <p>El sistema deberá alertar de forma sonora con el texto "Ha seleccionado el modo Manual'.</p>
Resultado de la Prueba	Éxito

Tabla 97.: Prueba PFUN – 01.

Identificador	PFUN – 02
Descripción	Prueba del servicio Google Translate.
Requisito Software	RSF – 30, RSF – 31
Estado inicial	Aplicación instalada en el dispositivo.
Estado Final	Aplicación en cualquiera de los dos modos de la aplicación mostrando interfaz de elementos.
Procedimiento de Prueba	<p>Ir a ajustes del dispositivo.</p> <p>Seleccionar la etiqueta 'Teclado e Idioma'.</p> <p>Seleccionar la etiqueta 'Seleccionar Idioma'.</p> <p>Seleccionar cualquier idioma distinto del español.</p> <p>Iniciar la aplicación.</p> <p>Chekear la opción de Manual.</p> <p>Pulsar el botón 'Go!' de la interfaz.</p> <p>El sistema deberá alertar de forma sonora con el texto "Ha seleccionado el modo Manual' en el idioma en el que se haya cambiado el dispositivo.</p>
Resultado de la Prueba	Éxito

Tabla 98.: Prueba PFUN – 02.

Identificador	PFUN – 03
Descripción	Prueba del servicio Google VoiceRecognizer.
Requisito Software	RSF – 15, RSF – 16
Estado inicial	Aplicación instalada en el dispositivo.
Estado Final	Aplicación en modo ‘automático’ mostrando los las acciones del elemento seleccionado.
Procedimiento de Prueba	<p>Iniciar la aplicación.</p> <p>Chekear la opción de Automático.</p> <p>Pulsar el botón ‘Go!’ de la interfaz.</p> <p>El sistema mostrará la interfaz de elementos y acto seguido pedirá que se diga un elemento.</p> <p>El ‘tester’ ha de decir un elemento del sistema y esperar a ver si el servicio de google voice recognizer funciona.</p>
Resultado de la Prueba	Éxito

Tabla 99.: Prueba PFUN – 03.

Identificador	PFUN – 04
Descripción	Prueba del servicio HTTP Request.
Requisito Software	RSF – 08, RSF – 28, RSF – 29, RSF – 34, RSF – 35, RSF – 36.
Estado inicial	Aplicación en cualquiera de los dos modos en la interfaz de acciones.
Estado Final	Aplicación reproduce de forma sonora el texto “Enhorabuena, el sistema se va a... comando enviado al servidor”.
Procedimiento de Prueba	<p>Iniciar la aplicación.</p> <p>Chekear la opción de Manual.</p> <p>Pulsar el botón ‘Go!’ de la interfaz.</p> <p>El sistema mostrará la interfaz de elementos, pulsar uno.</p> <p>El sistema mostrará la interfaz de acciones, pulsar una.</p>
Resultado de la Prueba	Éxito

Tabla 100.: Prueba PFUN – 04.

Identificador	PFUN – 05
Descripción	Prueba del servicio SQLite.
Requisito Software	RSF – 08, RSF – 09, RSF – 10, RSF – 11, RSF – 12, RSF – 13, RSF – 14
Estado inicial	Aplicación en la interfaz de ajustes.
Estado Final	Aplicación reproduce de forma sonora el texto “Enhorabuena, el sistema se va a... comando enviado al servidor”.
Procedimiento de Prueba	<p>Iniciar la aplicación.</p> <p>Chekear la opción de Manual.</p> <p>Pulsar el botón ‘Go!’ de la interfaz.</p> <p>El sistema mostrará la interfaz de elementos, pulsar uno.</p> <p>El sistema mostrará la interfaz de acciones, pulsar una.</p> <p>El sistema emitirá un sonido de confirmación cuando la url halla sido mandada. Por tanto ha relacionado todos los requisitos especificados.</p>
Resultado de la Prueba	Éxito

Tabla 101.: Prueba PFUN – 05.

9. Conclusiones y Líneas Futuras.

En este apartado, se comentarán una serie de conclusiones y líneas futuras de trabajo para próximas actualizaciones del producto.

En el caso de las conclusiones, ha resultado esencial concluir el proyecto para poder dar las valoraciones adecuadas al mismo. Además, debido a las dificultades surgidas a lo largo del desarrollo del proyecto, se basan en la experiencia de haberlos solucionado.

Finalmente las líneas futuras, se han debatido con el responsable de la empresta tecnológica de domótica Jorge Alba. Dado que el producto lo mantendrán y evolucionarán futuros desarrolladores, se ha querido marcar un poco el camino a seguir, desde la experiencia del desarrollo de la primera versión.

9.1 Conclusiones.

Las conclusiones obtenidas tras la finalización del proyecto, son las siguientes:

- El sistema trata de ser lo más parecido al sistema diseñado en primera instancia. Debido a una fase de análisis del sistema efectiva, se ha conseguido que así sea.
- El equipo de desarrollo ha tenido que incrementar sus conocimientos tanto en el sistema operativo Android, como en las condiciones que rodean una persona con discapacidad severa.
- Las API's de traducción gratuitas son muy poco eficientes y los únicos servicios viables son los de pago, Google Translate o Microsoft Bing. Siendo el primero, la mejor de las opciones.
- Las posibilidades que da Android, hacen que la usabilidad del sistema sea muy fluida y de gran capacidad.

Como se puede observar, las conclusiones abarcan la mayor parte del desarrollo del sistema. Se debe a que se trata el primer proyecto desarrollado para domótica asistencial en Android.

9.2 Líneas futuras de mejora.

El equipo de desarrollo de éste proyecto, ha decidido consensuar las líneas futuras de mejora con el equipo de la empresa de domótica asistencial. Los puntos sacados en claro de dicha reunión, son:

- Una primera línea de futuros desarrollos, puede ser la de integrar una API propia de traducción. A pesar de que se requieren una gran cantidad de recursos para poder desarrollar una API de traducción efectiva, merece la pena destacar esta mejora.
- Otra posible mejora puede ser la portabilidad de la aplicación a distintos sistemas operativos de dispositivos móviles. Se priorizará la portabilidad al sistema operativo iOS de Apple.
- Además, el sistema desarrollado para el dispositivo, podrá integrar un servicio que gestione ciertos recursos del dispositivo como puede ser la batería o simplemente el estado en general del dispositivo. Para ello, se integrarían funcionalidades en el servidor domótico según las cuales, por ejemplo, si llegase la señal de 'batería baja' al servidor éste debería avisar al administrador del sistema.
- Deshabilitar la selección de la acción ya realizada sobre el objeto. Por ejemplo, si una luz se encuentra encendida, ésta, no puede volver a ser encendida. Actualmente, el sistema indica que se ha producido un error y el servidor no ha realizado la acción. No obstante, otra posible implementación es la de la deshabilitación de dicho mandato.

Las líneas de mejora marcadas por el equipo de desarrollo se han basado en la experiencia de desarrollo del sistema inicial. Se recomienda de forma incondicional la sucesiva implantación de las mejoras aquí marcadas.

Anexo I – Manual de Usuario.

En este primer Anexo, se facilitará el manual de usuario del sistema. Se espera con ello, que solucione posibles problemas de usabilidad del usuario, así como mostrar el funcionamiento íntegro de la aplicación.

Introducción.

El manual de usuario pretende exponer de forma clara y directa el funcionamiento del sistema. Además, se pretenden exponer los distintos modos de la aplicación, tanto el modo manual como el automático.

El propósito adicional de este manual, es el de resolver posibles dudas por parte del usuario, de cara al uso del sistema. El primer paso para la ejecución de esta aplicación es, seleccionar el icono de la aplicación ya instalada en el dispositivo, el icono en sí, es el expuesto en la siguiente ilustración.



Ilustración 39: Icono de la aplicación a ejecutar.

Modo de Uso.

El modo de uso de esta aplicación se puede dividir en dos tipos. La primera, el modo “Manual”, la segunda, el modo “Automático”. De forma genérica, se ha de apuntar que para el inicio de la aplicación se ha de especificar qué modo se quiere seleccionar. Una vez seleccionado el modo y pulsado el botón ‘Go!’ se iniciará alguno de los siguiente modos.

En la ilustración siguiente, se exponen los pasos para iniciar la aplicación ya instalada en el terminal. Como se puede observar, basta con pulsar el icono de la aplicación de forma táctil.

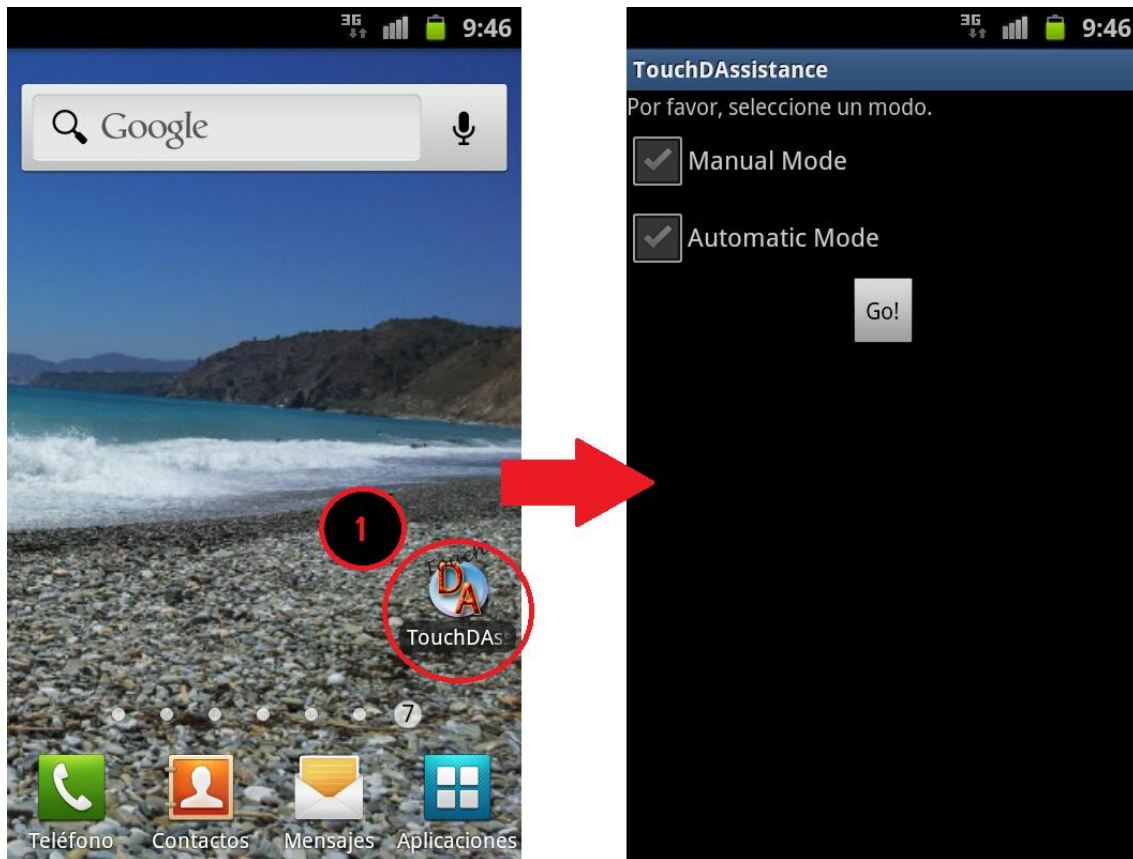


Ilustración 40: Captura de pantalla de inicio de la aplicación.

Modo automático.

Éste es el primer y principal modo de la aplicación. Se trata del modo automático adaptado para personas discapacitadas. En él, se realizará el reconocimiento por voz tanto de los elementos como de las acciones.

De forma automática, se realizará el reconocimiento de voz para el módulo de domótica asistencial. Como se puede observar en la ilustración expuesta a continuación el uso de la aplicación en éste modo se encuentra automatizado.

En la ilustración siguiente, se exponen los pasos para iniciar el modo 'automático'. Se puede observar que pulsando el 'checkbox' de "Automatic Mode"

y seguidamente el botón 'Go!' de la interfaz, se inicia el modo automático de la propia aplicación.



Ilustración 41: Captura de pantalla de la selección del modo automático de aplicación.

En la ilustración anterior, se muestra cómo al iniciar el modo de la aplicación automático, se inicia el reconocimiento de voz y se procesa el resultado.

En caso de que el elemento seleccionado exista en el sistema, se continuará a la siguiente interfaz en la que se muestran las acciones relacionadas con el elemento seleccionado. Como se muestra en la siguiente captura, el reconocimiento en esta interfaz también se encuentra automatizado.



Ilustración 42: Captura de pantalla de la selección de acción en modo automático de aplicación.

Finalmente, la aplicación mandará la acción procesada por el sistema al servidor domótico de la instalación. En caso de error en el procesamiento del comando de voz enviado, ésta, pedirá de nuevo el mandato indicado por voz.

Modo manual.

Éste es el modo adicional que se introduce en el sistema. Éste tipo de modo, se considerará válido únicamente para usuarios no impedidos. En él, se podrán seleccionar tanto los elementos como las acciones de forma manual por medio de la pantalla táctil.

Como se puede observar en las ilustraciones a continuación expuestas, el uso de la aplicación es bastante intuitivo y sencillo.



Ilustración 43: Captura de pantalla de la selección y uso del modo manual de la aplicación.

Como se puede observar en las ilustraciones anteriores, el sistema completo es utilizado mediante el uso de la pantalla táctil del dispositivo. Cuando finalice el proceso, se emitirá una alerta sonora en la que se dirá de forma textual si el mandato ha sido enviado al servidor domótico correctamente.

Además, señalar que en las Interfaces dos y tres, existe un botón 'Back' el cual nos permite navegar entre las distintas interfaces del sistema.

Anexo II – Manual de Instalación.

En el segundo de los anexos se va exponer el manual de instalación de la aplicación. En éste, se explicarán todos los detalles concernientes a la descarga e instalación de la aplicación.

Introducción.

Las aplicaciones móviles del sistema operativo Android, se encuentran disponible para la descarga desde el “Market” de Android, “Google Play”. Además cualquier dispositivo móvil de sistema operativo Android, con la simple aceptación de los términos de la aplicación, automatiza la descarga y la instalación de la misma.

Por lo que se consideran pasos sencillos, a pesar de ello, en este apartado se realizará una breve descripción de los pasos a seguir para una correcta instalación del sistema.

Pasos para la instalación.

En este apartado se realizará una breve descripción para todos y cada uno de los pasos de instalación de la aplicación móvil. Se detalla cómo encontrar la aplicación en Google Play. Además, se define la categoría y disponibilidad de la misma y finalmente los pasos para la descarga e instalación.

Market de Android.

El Market de Android o Google Play, como se ha renombrado actualmente es el repositorio de aplicaciones de éste sistema operativo. Todos los dispositivos con sistema operativo Android, han de tener una aplicación cuyo logo es el siguiente:

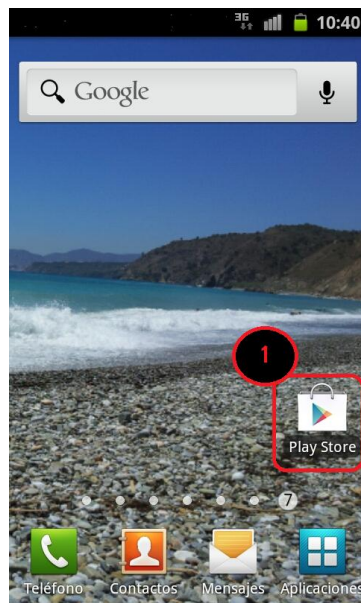



Ilustración 44: Captura de pantalla del icono de 'Play Store'.

Una vez seleccionado y ya dentro de esta aplicación, el usuario deberá buscar la aplicación. Para ello, se puede realizar de dos modos, el primero, introduciendo el nombre de la aplicación seleccionando la lupa  e introduciendo en el campo de texto el nombre de la misma "Domotic Assistance" La segunda forma, será ingresando en la categoría que se define más adelante.

Categoría y Disponibilidad.

Para buscar la aplicación según 'Categoría' se ha de seleccionar en primer lugar el apartado de "Aplicaciones", como se puede ver en la ilustración que se expone a continuación.



Ilustración 45: Captura de pantalla del botón 'Aplicaciones'.

Seguidamente, se ha de deslizar el menú superior hacia la izquierda de tal modo que aparezca el menú de 'Categorías' como se puede observar en la siguiente ilustración, dicho menú contiene múltiples categorías, deslizando hacia abajo las opciones encontraremos la que queremos.



Ilustración 46: Captura de pantalla del menú 'Categoría'.

Una vez que el usuario se encuentre en dicho menú, ha de seleccionar la categoría de 'Productividad'. En dicho apartado se podrá realizar una búsqueda más refinada de nuestra aplicación.

Descarga e Instalación.

Una vez encontrada nuestra aplicación, se ha de aceptar seleccionando el botón que se muestra en la ilustración a continuación.

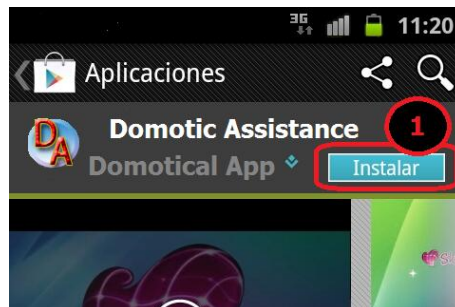


Ilustración 47: Captura de pantalla del botón 'Instalar'.

Seguidamente, nos aparecerá una interfaz en la cual se deberá seleccionar el botón 'Aceptar y Descargar' el cual se muestra a continuación. En esta interfaz, se mostrarán los permisos de los que requiere la aplicación.

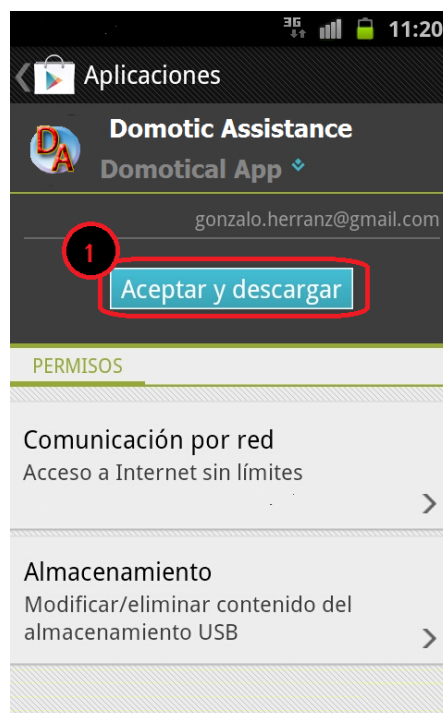


Ilustración 48: Captura de pantalla del botón 'Aceptar y Descargar'.

Anexo III – Acrónimos, definiciones y referencias.

En este último anexo, se expondrán una serie de definiciones acrónimos y referencias básicas para seguir la lectura del documento. Con él, se pretende aclarar las dudas que se puedan generar a medida que se produzca la lectura del documento. Se tratará de identificar todos los términos técnicos o tecnicismos, que puedan no ser reconocidos por un lector no especializado.

Acrónimos y definiciones.

En este apartado se especificarán en primer lugar los acrónimos que han aparecido en el documento. Seguidamente, se expondrán todas las definiciones de los términos que puedan ser de difuso entendimiento, para ello se recurrirá a definiciones lo más simples y claras posibles. Se recurrirá en todas ellas al recurso de enciclopedia online, wikipedia^[4].

Acrónimos.

API – Application Programming Interface.

ASI – Análisis del Sistema de Información.

BBDD – Bases de Datos.

CeDINT – Centro de Domótica Integral.

CENELEC – Comité Europeo de Normalización Electrotécnica.

CSI – Construcción del Sistema de Información.

CSIC – Centro de Science Investigación.

DSI – Diseño del Sistema de Información

FAE – Fundación de Alzheimer de España.

HTTP – HyperText Transfer Protocol

IAS – Implantación y Aceptación del Sistema.

ISO – International Organization for Standardization.

ITADA – Interfaces Tangibles Avanzadas para Domótica Asistencial.

POSIX – Portable Operating System Interface.

SMS – Short Message System.

SSOO – Sistemas Operativos

UML – Unified Modeling Language.

Definiciones.

Android – Sistema operativo móvil comprado por Google a la empresa informática, 'Android Inc'. Está basado en Linux, que junto con aplicaciones middleware, está enfocado para ser utilizado en móviles inteligentes, tabletas informáticas...

API – Se trata de la especificación que se ha de usar como una interfaz por algunos componentes del sistema software. Una API, debe incluir múltiples elementos, como por ejemplo llamadas POSIX o documentación de la misma.

Apps – Abreviatura de la palabra 'aplicaciones' normalmente se refiere a software informático, más concretamente a las aplicaciones de los sistemas móviles.

Bus – En la jerga informática, un 'BUS' es un sistema digital que transfiere datos entre los componentes de una computadora o distintas computadoras. Está formado por cables o pistas en un circuito impreso.

Framework – Espacio de trabajo destinado al programador. Su utilidad, desarrollar aplicaciones en cualquier lenguaje de programación soportado por el mismo.

Gestos Multi-touch – Se trata de la técnica de interacción persona-computador y al hardware que le aplica. La tecnología 'Multi-Touch' de Apple, consiste en una pantalla táctil que reconoce simultáneamente distintos puntos de contacto.

Google Accesories – Se trata de un servicio de Google que integra gran parte de sus habituales servicios.

Google Play ó Market – Se trata de una tienda de software en línea desarrollada por Google para los dispositivos 'Android'. Es una aplicación que está preinstalada en la mayoría de los dispositivos con el mencionado sistema operativo.

Google Translate – Es un sistema de traducción automática del cual es propietario 'Google'. Está basado en datos estadísticos para traducir texto de documentos y páginas web.

Google Voice Recognizer – Se trata de un servicio ofrecido por google de forma gratuita, según la cual, a partir de la frecuencia de una voz, se reconoce el mandato que está diciendo el usuario.

HTML5 – Es la versión quinta de revisión del lenguaje básico del World Wild Web, HTML. Éste lenguaje, especifica dos variantes de sintaxis para HTML, el tipo text/html y el tipo XHTML5.

HTTP Request – Se trata de una de las funciones del protocolo HTTP.

iPhone – Es una familia de teléfonos inteligentes multimedia con conexión a Internet, pantalla táctil capacitiva y escasos botones físicos. Diseñado por Apple Inc. El primero, desarrollado en el año 2007.

Java – Lenguaje de programación de alto nivel orientado a objetos, desarrollado por James Gosling en 1995. Obtiene mucha sintaxis de distintos lenguajes predecesores, no obstante tiene un modelo de objetos mucho más simple.

KNX Tecnologías – Se trata del estándar utilizado para la instalación del sistema domótico descrito.

Open Source – Concepto informático que implica que el software no está bajo la licencia de ningún propietario.

Samsung Galaxy SII – Es un teléfono inteligente con sistema operativo 'Android' lanzado en el año 2011, contiene un procesador de 1,2GHz de doble núcleo y especificaciones técnicas muy avanzadas.

SmartPhone – Es un teléfono móvil construido sobre una plataforma informática móvil. Con gran capacidad de computación y conectividad que un teléfono móvil convencional.

SQLite – Se trata del sistema de gestión de bases de datos relacional compatible con ACID. Se trata de un proyecto de dominio público. Su gran versatilidad, y escalabilidad fueron factores claves en la introducción al sistema operativo Android.

TextToSpeech – Se trata del servicio de conversión texto-voz. La generación de voz artificial que genera un sonido idéntico al que produciría una persona al leer un texto.

Viber – Aplicación para dispositivo móvil disponible para varios sistemas, como Android, iOS o BlackBerry OS. Entre sus funciones se encuentran la de establecer llamadas vía VoIP.

VoIP – Es un grupo de recursos que hacen posible que la señal de voz viaje a través de Internet empleando un protocolo IP. Esto significa que se envía la señal de voz en forma digital.

WhatsApp – Software propietario multiplataforma de mensajería instantánea entre teléfonos inteligentes. Además del envío de texto, permite la transmisión de imágenes, vídeo y audio.

Referencias.

En este apartado se expondrán las referencias que han ayudado en cierto grado a la redacción de este documento. En primer lugar, se expondrá una breve descripción de la ayuda prestada, y seguidamente se facilitará la dirección web de la misma.

1. Ayudas a la dependencia:

http://www.enfermeriasalamanca.com/legislacion/Ley_de_dependencia_resumen.pdf

2. Métrica v.3, ministerio de España.

http://administracionelectronica.gob.es/?_nfpb=true&_pageLabel=P60085901274201580632&langPae=es

3. Android, página oficial de desarrolladores.

<http://developer.android.com/index.html>

4. Enciclopedia Virtual. Wikipedia.

<http://www.wikipedia.com>

5. KNX, empresa de domótica:

<http://www.knx.org>

6. API del servicio ‘Google Translate’:

<https://developers.google.com/translate/>

7. API de Java:

<http://docs.oracle.com/javase/6/docs/api/>

